AD________________

AWARD NUMBER: DAMD17-03-1-0520

TITLE: A System for Discovering Bioengineered Threats by Knowledge Base Driven Mining of Toxin Data

PRINCIPAL INVESTIGATOR: S. Swaminathan, Ph.D.

CONTRACTING ORGANIZATION: Brookhaven National Laboratories
Upton, New York 11973

REPORT DATE: August 2006

TYPE OF REPORT: Final

PREPARED FOR: U.S. Army Medical Research and Materiel Command
Fort Detrick, Maryland 21702-5012

DISTRIBUTION STATEMENT: Approved for Public Release;
Distribution Unlimited

The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision unless so designated by other documentation.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE *(DD-MM-YYYY)* 01-08-2006	**2. REPORT TYPE** Final	**3. DATES COVERED** *(From - To)* 1 Aug 2003 – 31 Jul 2006

4. TITLE AND SUBTITLE

A System for Discovering Bioengineered Threats by Knowledge Base Driven Mining of Toxin Data

5a. CONTRACT NUMBER

5b. GRANT NUMBER
DAMD17-03-1-0520

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

S. Swaminathan, Ph.D.

E-Mail: swami@bnl.gov

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Brookhaven National Laboratories
Upton, New York 11973

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Medical Research and Materiel Command
Fort Detrick, Maryland 21702-5012

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This project was funded to establish a Toxin Knowledge Base (TKB) which will encompass information about bacterial toxins in general and toxins relevant to biodefense, in particular. The overall goal of this project is to establish an easy to use database viz. a Knowledge Base to populate itself and expand using machine learning techniques, to make it more dynamic. It is designed to be a bioinformatics resource focused on molecular information about toxins and other virulence factors that are the natural products of biological and potential biological warfare (BW and PBW) agents. The major aim was to mine, assimilate, synthesize, analyze and disseminate genomic and structural information on BW and PBW genes and their products. Using advanced machine learning and data mining the TKB has been developed to look for motifs, to design new experiments and also to predict structure and function of molecules (including putative chimeras) for which these data are not available. TKB will use innovative computer methods to parse the literature available in public resources (web sites) to identify new and emerging toxins to be included in the database.

15. SUBJECT TERMS
Toxin data base, toxin homologs, bioengineered threat, text mining, HMM profile

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** USAMRMC
a. REPORT U	**b. ABSTRACT** U	**c. THIS PAGE** U	UU	52	**19b. TELEPHONE NUMBER** *(include area code)*

Table of Contents

A System for Discovering Bioengineered Threats by Knowledge Base Driven Mining of Toxin Data

Introduction

This project was funded to establish a Toxin Knowledge Base (TKB) which will encompass information about bacterial toxins in general and toxins relevant to biodefense, in particular. This report includes the work done in the no-cost extension period as well. The overall goal of this project is to establish an easy to use database *viz*. a Knowledge Base to populate itself and expand using machine learning techniques, to make it more dynamic. It is designed to be a bioinformatics resource focused on molecular information about toxins and other virulence factors that are the natural products of biological and potential biological warfare (BW and PBW) agents. The major aim was to mine, assimilate, synthesize, analyze and disseminate genomic and structural information on BW and PBW genes and their products. Using advanced machine learning and data mining the TKB has been developed to look for motifs, to design new experiments and also to predict structure and function of molecules (including putative chimeras) for which these data are not available. TKB will use innovative computer methods to parse the literature available in public resources (web sites) to identify new and emerging toxins to be included in the database.

Body

A. Design and implementation of a highly curated Toxin Knowledge Base:

During the project period we have modified, improved and expanded the previously existing database for storing, managing and accessing molecular information on known as well as potential biological toxins. Here we are presenting a complete description with examples. This section describes our work, progress and deliverables as given in Specific Tasks 1, 2 and 3. As of now, the system is ready to be released as alpha version to test the reliability and usefulness by interested scientists. Accordingly, we are requesting permission from the Army to open this as a public website with safeguards and security in place. We will develop proper procedures to keep the data and the site secure.

System Architecture

The primary motivation for developing TKB was to address the need to establish an infrastructure resource that will aid studies in (1) developing methods for identifying potential bio-warfare agents, (2) identifying and developing counter measures such as anti-toxins, vaccines, and inhibitors, and (3) developing a better understanding of the mode of actions of these toxins at the cellular, sub-cellular, and molecular levels. TKB also focuses on correlating known and predicted 3-dimensional structures for these toxins with sequence, function, and biological activity. In order to develop a system that satisfies all these aims, we have developed a comprehensive architecture that accommodates the needs of a growing system.

TKB is comprised of two major components: (1) A powerful data-acquisition/ administration system for direct deposition of data related to toxins and (2) an ad-hoc query and reasoning system to access and to analyze information. Figure 1 shows the system architecture of TKB showing the querying and reasoning subsystem and the data acquisition subsystem. It also shows the architecture of the system, from the users' perspective.

Toxin Knowledge Base (TKB) is used to store biological information about various kinds of toxins. It stores homologs and active site information for each toxin and models for the homologs. It provides two interfaces to the user namely:

1. **Query and Reasoning Interfaces**: This facilitates the following:
    a. Toxin Search - Selective retrieval of toxin information.
    b. Homology Search - Finding toxins that are homologous to a given protein sequence.
    c. MuToxin - Determining whether a protein can be transformed into a toxin.
2. **Administrative Interfaces**: This interface is accessible only to a user with administrative rights. The toxin knowledge base can be updated in two ways:
    a. User-initiated: This involves updating the knowledge base with newly identified toxin information and related active site information.

b. Automated: This involves updating the knowledge base with new homologs and their models for the toxins on a periodic basis, so as to keep the toxin knowledge base up-to-date.
c. User Approval: This allows a new user's identity to be verified and approved for use of the TKB.

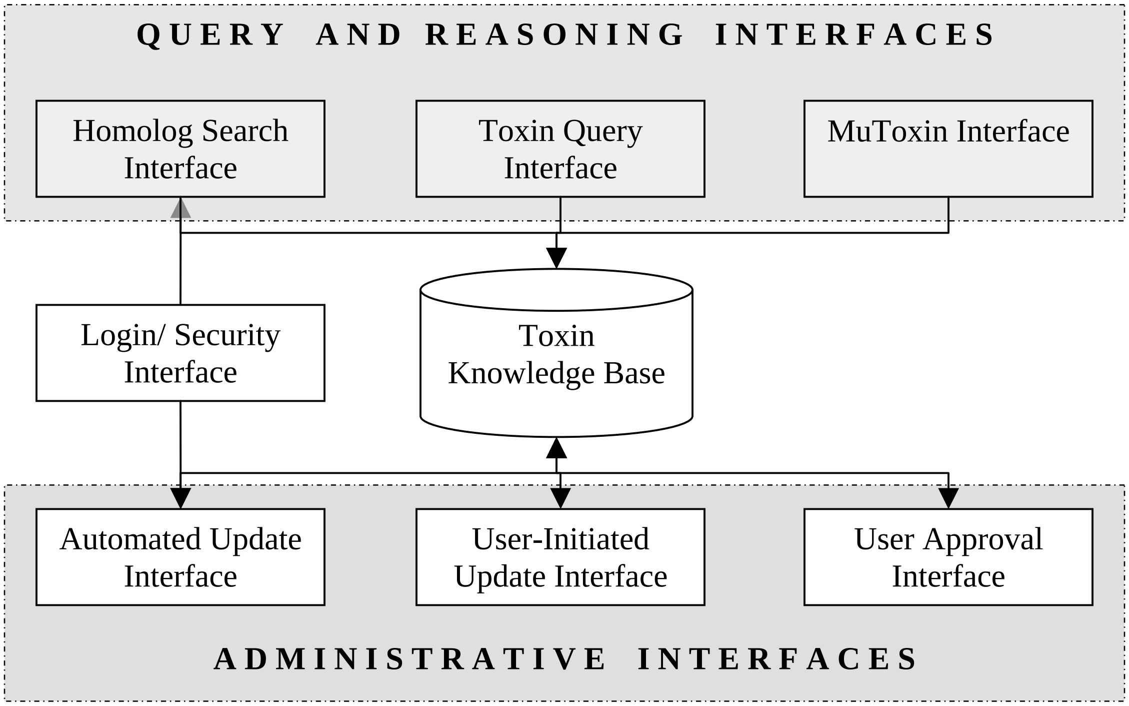

QUERY AND REASONING INTERFACES
Homolog Search Interface
Toxin Query Interface
MuToxin Interface
Login/ Security Interface
Toxin Knowledge Base
Automated Update Interface
User-Initiated Update Interface
User Approval Interface
ADMINISTRATIVE INTERFACES

Figure 1: System Architecture: The above figure represents a concise architecture of the system which has been developed by Brookhaven National Laboratory and Stony Brook University. The toxin knowledge base essentially is a data source which provides two kinds of interfaces to the user – one used to query the knowledge base and the other used to update the information in the knowledge base.

TKB integrates several publicly available tools that were developed for various unrelated purposes, but which are engineered into a workflow for identifying potential mutated toxins. This is a part of the query and reasoning interface, and is explained further in the section on the Query and Reasoning Interfaces. The system's architecture and description are organized as follows. First, the powerful data acquisition system is presented, along with the workflow used for the same. Second, we present the logical query and inference system that has been developed to identify a protein that can be converted into a toxin. The implementation details of the system is finally presented, with a report on the current status of the knowledge base, and followed by a section on the results obtained so far using the system.

Data Acquisition System

Public databases of biological information are popular research tools in the biological community. While providing wealth of information, they offer little help in analyzing, assimilating, and collecting data related to a particular topic (like toxins). As a result, the user is forced to search through multiple data sources and correlate the data manually. TKB fills a sorely needed gap. In particular it is an integrated tool for collecting, aggregating, and analyzing toxin data from different *data sources*. The sources that we currently use in our data acquisition process are PUBMED, SWISS-PROT, and RCSB.

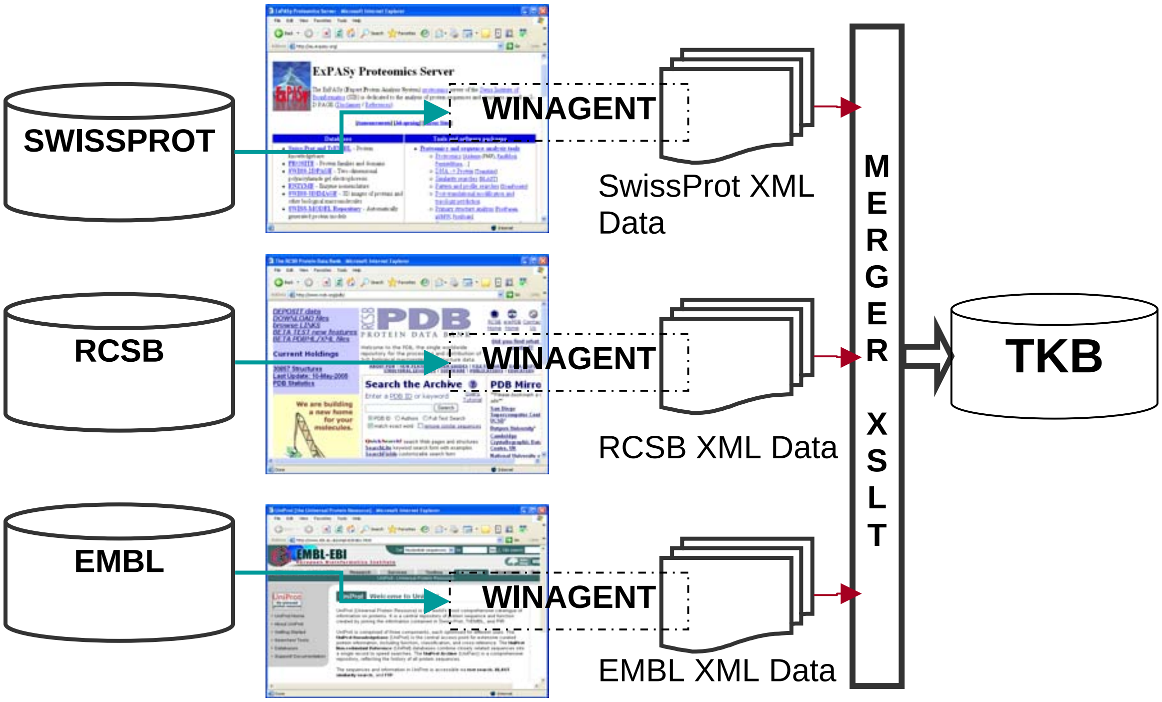


Figure 2: Data Acquisition Workflow. The data acquisition process consists of integrating data from disparate sources (publicly available bioinformatics databases) into a common data repository such that the information can be collected and assembled so that it can be queried.

In order to acquire data from vastly disparate sources like the RCSB, SWISSPROT and EMBL, an information extraction tool was built using an inbuilt tool known as WinAgent. This tool can be used to mine data from various web data sources. WinAgent is a software robot that learns to extract data from the Web by observing a user's navigation activity. By training the agent on the websites of interest, the user can easily teach the tool to acquire relevant data. In order to overcome the problem of data incompatibility among the different sites, a merger XSLT tool was built that compiles all the data into a single unified schema (explained further in this section) and stores it into the TKB. Most of the data acquisition is automated, except for situations in which a new data source has been identified, and the WinAgent has to be trained to extract information from such a data source. This process is also made easy by the fact that the training process is just a few clicks on the mouse and showing how the user would want to navigate the new data source. The data acquisition system includes algorithms that make it scalable in lieu of the rapidly growing amounts of data.

The data thus acquired is stored in a single unified schema, shown in figure 3. The schema has been developed with great care in order to include all relevant information a user might want to learn about a toxin. The sample schema shown here includes the details presented when the user looks at a particular toxin. In this example we present the schema when the user has selected botulinum neurotoxin type E as his choice. As shown in the figure the list of schema headings (left hand side of the table) is quite comprehensive. Certain fields have more information – providing a summary on the activity and mechanism of reaction (if it is available in the data sources or through literature search) of a toxin, where as certain fields do not have a lot of information, but represented using a hyperlink. This indicates that the information was collected from a different public source and can be then obtained by clicking on that relevant hyperlink.

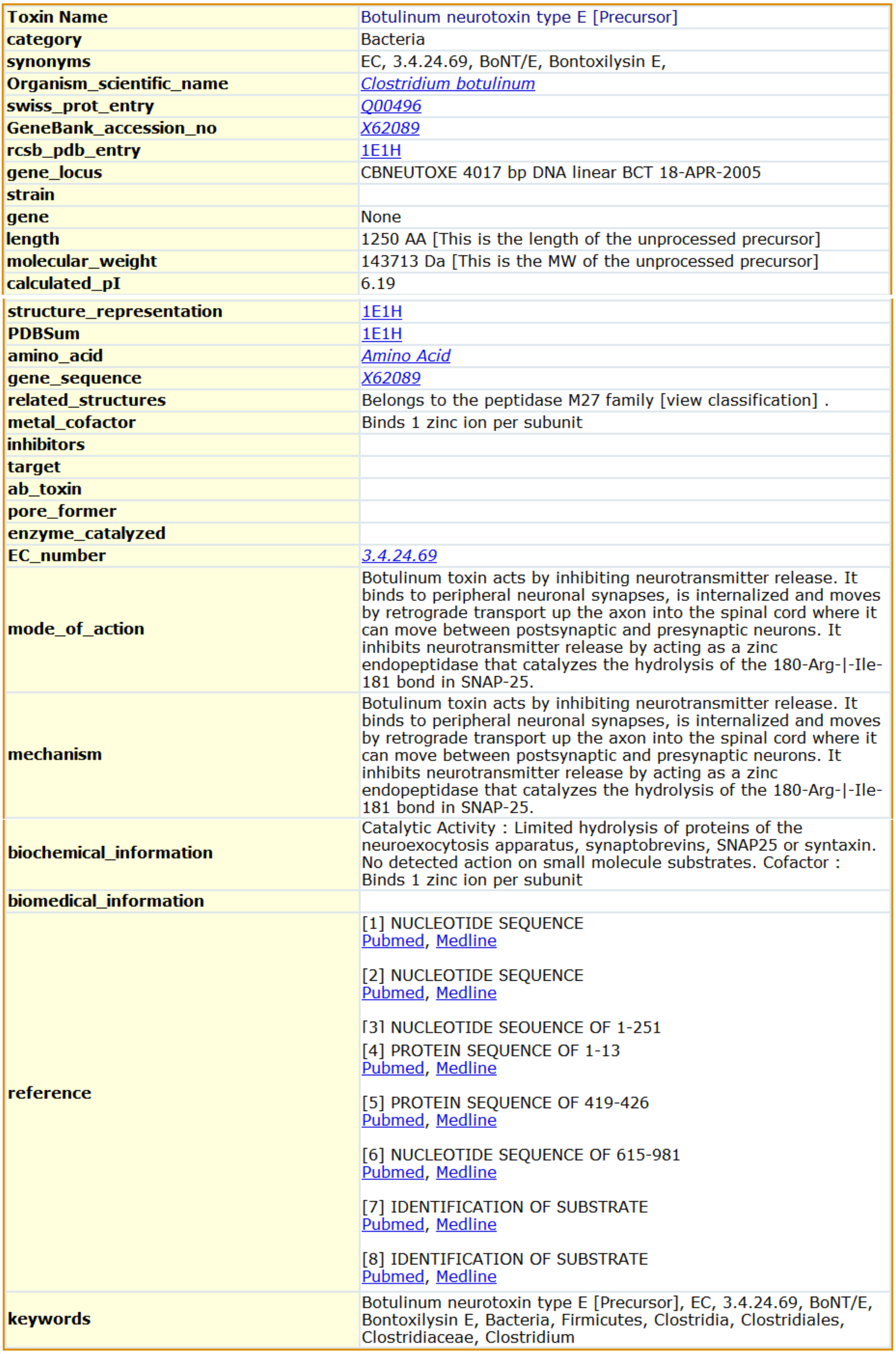
An exception to this case is the capability of the system to provide the user with the structure information, if it is available. Using an internet plug-in called Chime (http://www.mdli.com/chime), the user can directly link to the structure and perform various operations on the 3 dimensional structures using mouse buttons and key board buttons. This allows for increased interactivity with the system, and hence improves the overall experience for the user while using the TKB. This usage has been further shown in the usage section of the report.

The MuToxin (Trans-toxin) Workflow

A critical part of the system is dedicated to the derivation of new knowledge from the existing knowledge. Hence as part of the powerful query and reasoning system, we have included an engineered workflow that allows the end user to determine whether a given protein, (1) resembles a toxin at its active site and (2) whether residue substitutions at specific locations on the protein, can modify the protein into a toxin (minimally the active site). A flowchart is given in Figure 4.

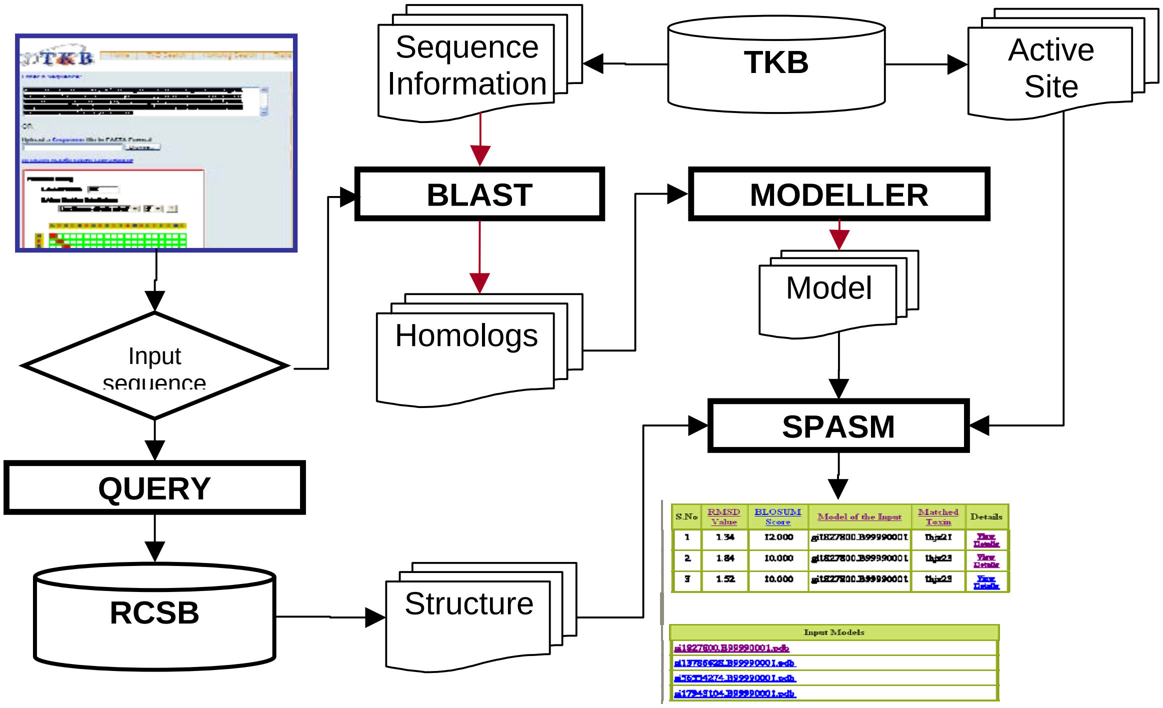
As shown in figure 4, the workflow integrates three separate off-the shelf bioinformatics and structural biology resources into a neat workflow. When the user provides an input protein sequence through the user interface, the homologs of the input sequence is collected and based on the homologs, if a structure exists within our structure database, a reasonable model is built using the Modeller program. Based on the active site information available, it is then provided as input to the SPASM program, which superposes the built model against a database of active site templates and compares them for some match using a customized substitution matrix score. This provides the end user with a reasonable estimate as to whether the input protein resembles a toxin in some fashion.

Another output from the workflow is a table of substitution scores and positions at which possible residue substitutions need to be made such that the active site resembles the target toxin. This provides information whether the protein can be a potential chimera (to hide a potentially toxic active site into a benign protein). All these are illustrated in the usage section of this report.



Toxin Name	Botulinum neurotoxin type E [Precursor]
category	Bacteria
synonyms	EC, 3.4.24.69, BoNT/E, Bontoxilysin E,
Organism_scientific_name	*Clostridium botulinum*
swiss_prot_entry	*Q00496*
GeneBank_accession_no	*X62089*
rcsb_pdb_entry	1E1H
gene_locus	CBNEUTOXE 4017 bp DNA linear BCT 18-APR-2005
strain	
gene	None
length	1250 AA [This is the length of the unprocessed precursor]
molecular_weight	143713 Da [This is the MW of the unprocessed precursor]
calculated_pI	6.19
structure_representation	1E1H
PDBSum	1E1H
amino_acid	*Amino Acid*
gene_sequence	*X62089*
related_structures	Belongs to the peptidase M27 family [view classification] .
metal_cofactor	Binds 1 zinc ion per subunit
inhibitors	
target	
ab_toxin	
pore_former	
enzyme_catalyzed	
EC_number	*3.4.24.69*
mode_of_action	Botulinum toxin acts by inhibiting neurotransmitter release. It binds to peripheral neuronal synapses, is internalized and moves by retrograde transport up the axon into the spinal cord where it can move between postsynaptic and presynaptic neurons. It inhibits neurotransmitter release by acting as a zinc endopeptidase that catalyzes the hydrolysis of the 180-Arg-|-Ile-181 bond in SNAP-25.
mechanism	Botulinum toxin acts by inhibiting neurotransmitter release. It binds to peripheral neuronal synapses, is internalized and moves by retrograde transport up the axon into the spinal cord where it can move between postsynaptic and presynaptic neurons. It inhibits neurotransmitter release by acting as a zinc endopeptidase that catalyzes the hydrolysis of the 180-Arg-|-Ile-181 bond in SNAP-25.
biochemical_information	Catalytic Activity : Limited hydrolysis of proteins of the neuroexocytosis apparatus, synaptobrevins, SNAP25 or syntaxin. No detected action on small molecule substrates. Cofactor : Binds 1 zinc ion per subunit
biomedical_information	
reference	[1] NUCLEOTIDE SEQUENCE Pubmed, Medline [2] NUCLEOTIDE SEQUENCE Pubmed, Medline [3] NUCLEOTIDE SEQUENCE OF 1-251 [4] PROTEIN SEQUENCE OF 1-13 Pubmed, Medline [5] PROTEIN SEQUENCE OF 419-426 Pubmed, Medline [6] NUCLEOTIDE SEQUENCE OF 615-981 Pubmed, Medline [7] IDENTIFICATION OF SUBSTRATE Pubmed, Medline [8] IDENTIFICATION OF SUBSTRATE Pubmed, Medline
keywords	Botulinum neurotoxin type E [Precursor], EC, 3.4.24.69, BoNT/E, Bontoxilysin E, Bacteria, Firmicutes, Clostridia, Clostridiales, Clostridiaceae, Clostridium

Figure 3: Overall schema representation for Botulinum neurotoxin type E [precursor]



S.No	RMSD Value	BLOSUM Score	Model of the Input	Matched Toxin	Details
1	1.34	12.000	gi1827800.B99990001	1hjz21	View Details
2	1.84	10.000	gi1827800.B99990001	1hjz23	View Details
3	1.52	10.000	gi1827800.B99990001	1hjz23	View Details

Input Models
gi1827800.B99990001.pdb
gi13786628.B99990001.pdb
gi56334274.B99990001.pdb
gi17949104.B99990001.pdb

Figure 4: The Mu-Toxin (Trans-Toxin) Workflow.

System Implementation

The system has been developed entirely using Java, Java Server Pages (JSP), HTML and XML/ XSLT technologies. Essentially organized into three layers (based on the Model View Controller design pattern), the front end (view) of the system consists of interacting JSP which are kept extremely functional. All the aspects of user views and definitions are made using XSLT, which allows for a very flexible front end to be developed. In fact the user is usually unaware of the existence of the XSLT since the code generated on the front end is very dynamic in nature.

The controller objects are developed as Java Servlets, with ability to handle multiple sessions, control opening and closing of new windows as and when required, pass session control to JSP and retain information for further processing of user commands. The controller objects do not generate any HTML artifacts except for some administrative logs that are stored at the server end for monitoring the status of the system.

The model (back end) is implemented using Oracle 10G as the primary database, with extensive support using XML. The database schema is very flexible in order to accommodate periodic changes that may be necessary because of the ever-expanding knowledge within the field of toxicology.

We also provide here the current status of the database and the various statistics as an estimate of the size of the database tables.

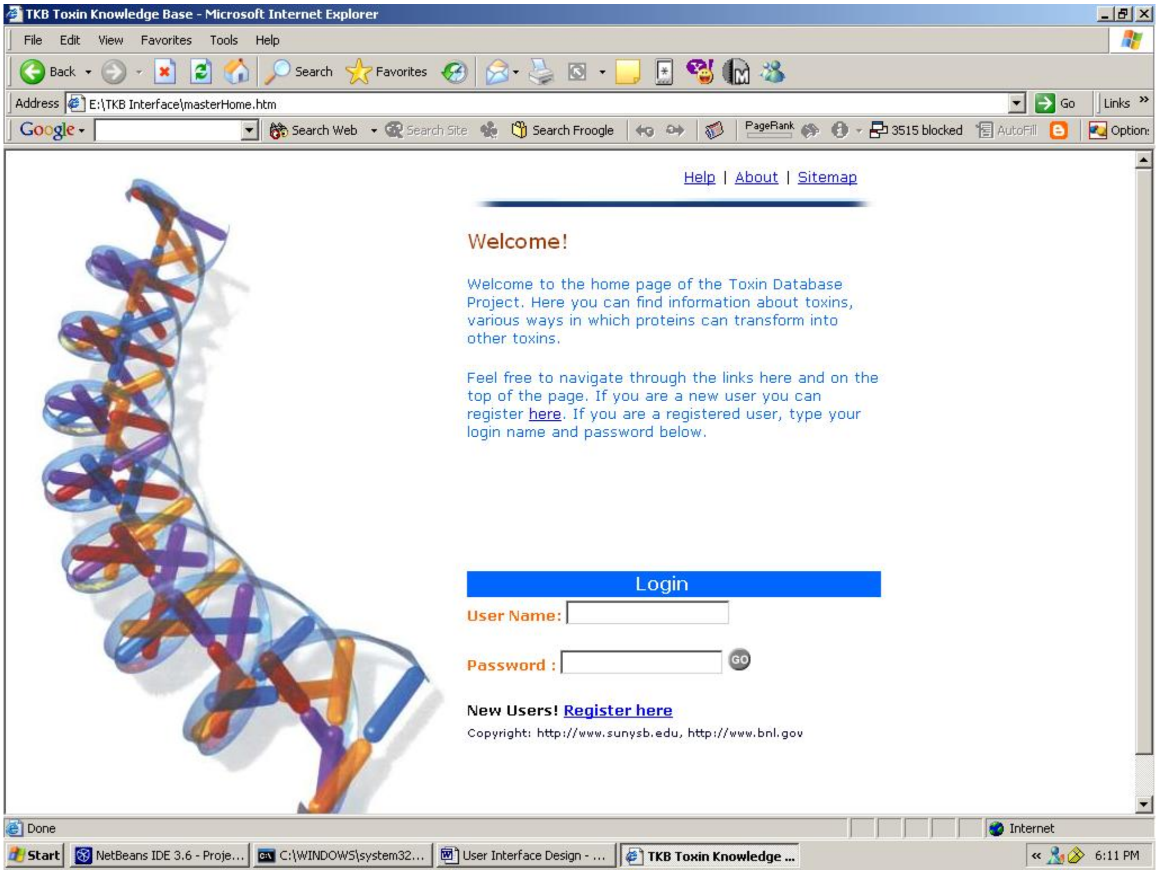
Total Number of Toxins	>1009
Number of Toxins with Structures	>539
Total number of Homologs	>79,658 (79 Homologs / Toxin)
Total size of Toxin database	~1.64 GB
Total number of indices used	~14

Usage Scenarios

In this section a detailed look into the system implementation is provided. We provide information about the various interfaces, how they are organized, how a typical user will navigate and use the system (both a normal user and an administrator) and also various screen shots of the system.

Logging into the System

An important step towards using the system is to have some form of authentication of the end users, so that the system is not compromised. To this effect all users need to log into the system. The login interface is a simple authentication mechanism, which verifies the user (through a suitable user-name/ password) and also provides access to the user interface that a user is privileged to use (Figure 5). This means that a user just needs to type in his user name and password – and the system then recognizes the privileges of the user and allows access to only those pages that he/she can use. This improves the security of the system by providing a single point of entry into the system.


TKB Toxin Knowledge Base - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address E:\TKB Interface\masterHome.htm
Go Links
Google Search Web Search Site Search Froogle PageRank 3515 blocked AutoFill Options
Help | About | Sitemap
Login
User Name:
Password :
GO
Done
Internet
Start NetBeans IDE 3.6 - Proje... C:\WINDOWS\system32... User Interface Design - ... TKB Toxin Knowledge ...
6:11 PM

Welcome!

Welcome to the home page of the Toxin Database Project. Here you can find information about toxins, various ways in which proteins can transform into other toxins.

Feel free to navigate through the links here and on the top of the page. If you are a new user you can register here. If you are a registered user, type your login name and password below.

New Users! Register here

Copyright: http://www.sunysb.edu, http://www.bnl.gov

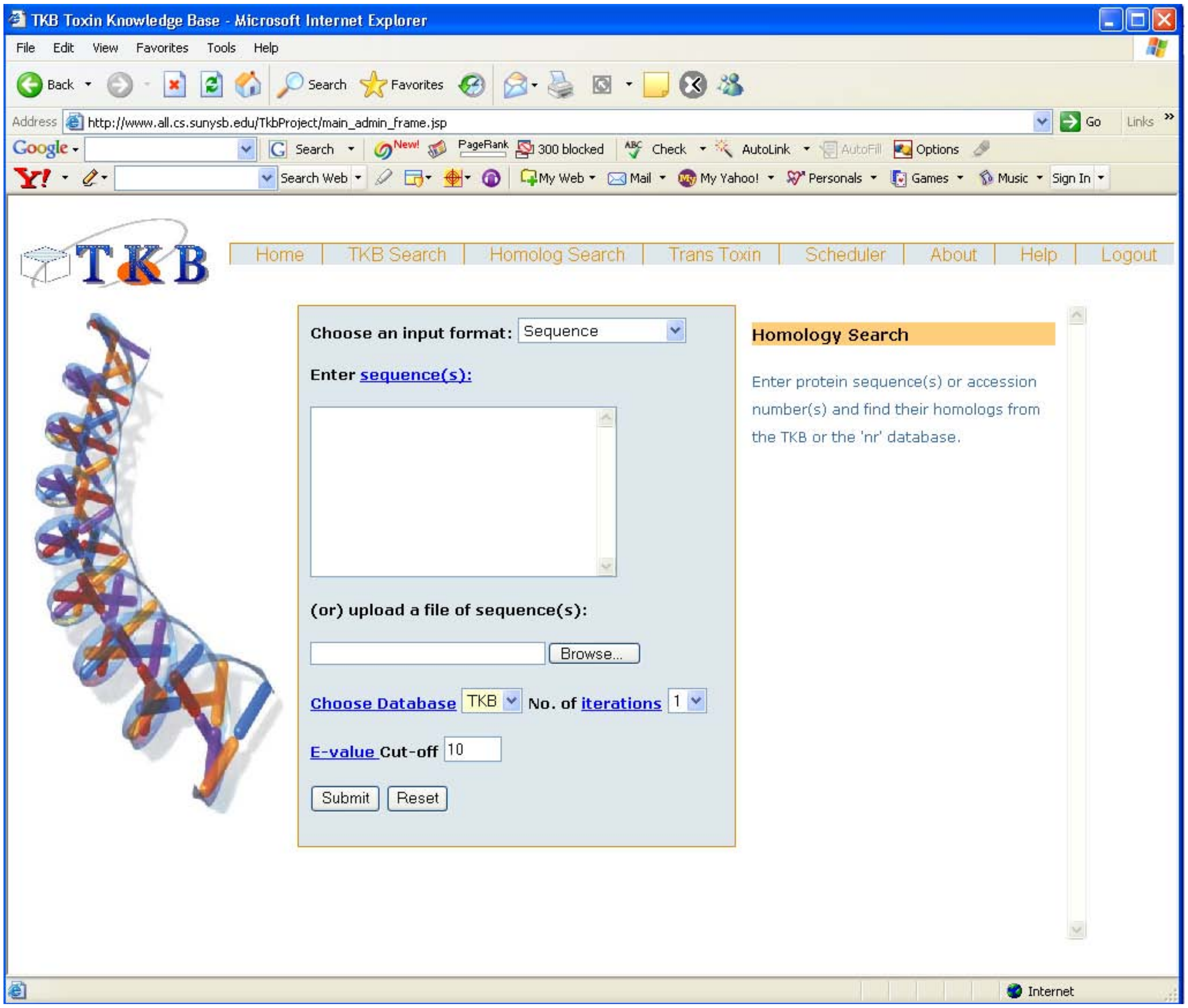
Figure 5: Login Screen: The login screen is a simple welcome page that allows the user to enter his user name and password so that he can login to the system. There is a separate form for new users who want to register and start using the system. However, new users can register only after they have been screened for security purposes.

Query and Reasoning Interface

The query interface provides facilities to the user to query and use the information stored in the knowledge base in different ways, described in the following sections.

Homology Search

- This interface helps in finding homologs of a given protein sequence from the TKB using PSI-BLAST (Position specific iterative BLAST), which is the NCBI tool for homology search.
- The "Homology search" interface (as shown below in Figure 6) accepts two forms of input from the user:
    1. A protein FASTA sequence (or a list of such sequences)
    2. An accession number (or a list of accession numbers)


TKB Toxin Knowledge Base - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address
Go Links
Google Search PageRank 300 blocked Check AutoLink AutoFill Options
Search Web My Web Mail My Yahoo! Personals Games Music Sign In
TKB
Home TKB Search Homolog Search Trans Toxin Scheduler About Help Logout
Browse...
Submit Reset
Internet

Address: http://www.all.cs.sunysb.edu/TkbProject/main_admin_frame.jsp

Choose an input format: Sequence

Enter sequence(s):

(or) upload a file of sequence(s):

Choose Database TKB No. of iterations 1

E-value Cut-off 10

Homology Search

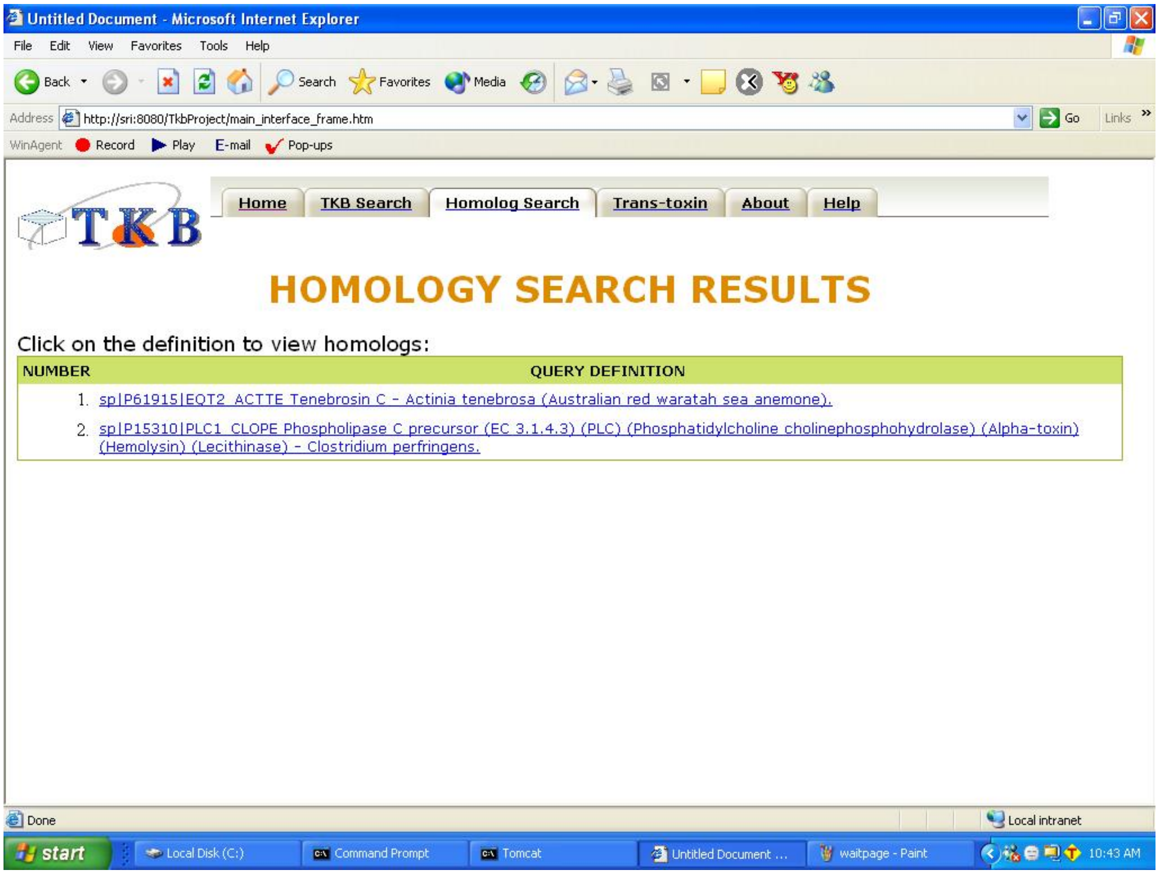
Enter protein sequence(s) or accession number(s) and find their homologs from the TKB or the 'nr' database.

Figure 6: Homolog Search Interface: The search interface helps user to find a homolog given either a sequence or an accession number. The right hand pane shows how a help page can be dynamically loaded based on the links on the form. This allows all users with minimal knowledge of the system to understand the terms and use the system with minimal training.

- The following are the options provided for the homology search:

  1. Database: Provides a choice of database to be BLASTed against. The two options that are currently provided are the “TKB” and “nr”(non-redundant database)
  2. Number of iterations: PSI-BLAST uses the results of each "iteration" to refine the profile. This iterative searching strategy results in increased sensitivity.
  3. E-value cut-off: Lets the user define the “expect” threshold for the homology search.

- When the user submits the required inputs and options, PSI-BLAST is used to search against the specified database and the results are presented to the user in a concise yet comprehensive fashion, as the Figure 7 depicts.



Address http://sri:8080/TkbProject/main_interface_frame.htm

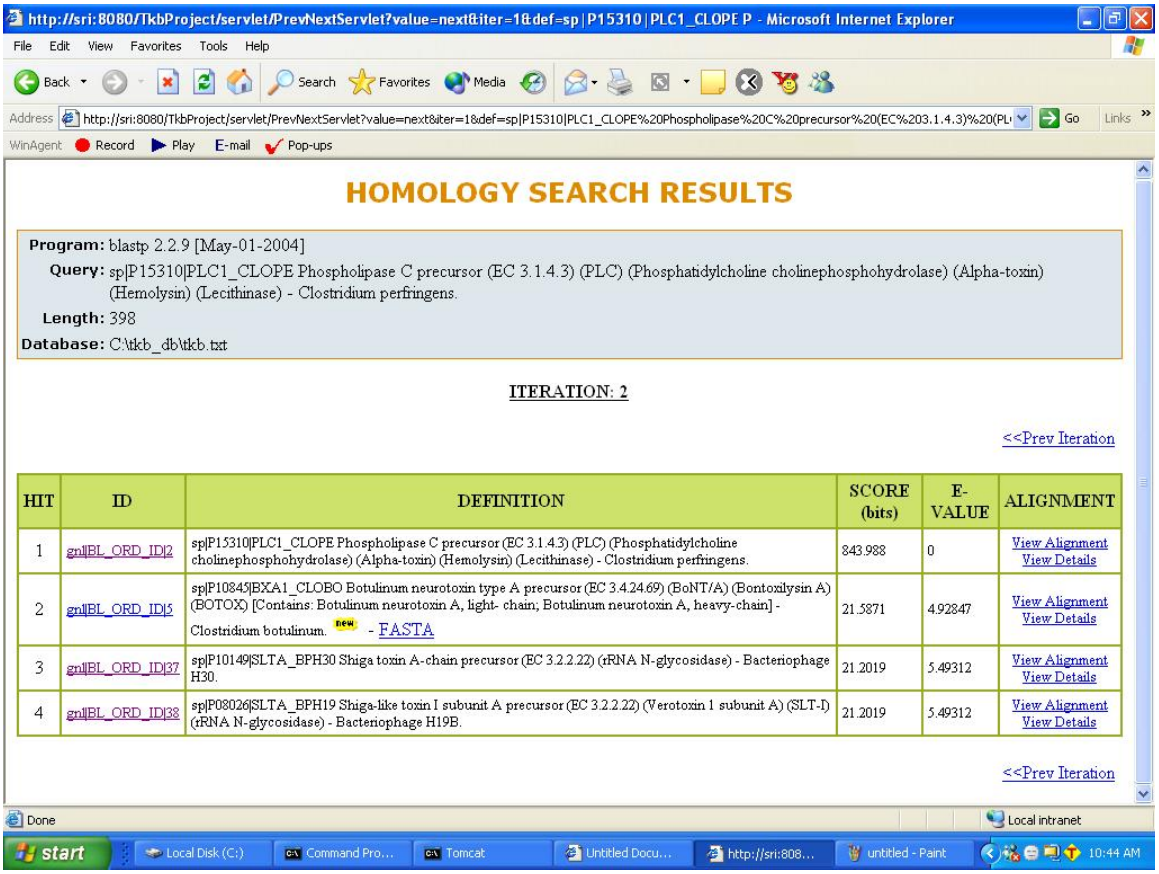
HOMOLOGY SEARCH RESULTS

Click on the definition to view homologs:

NUMBER	QUERY DEFINITION
1.	sp|P61915|EQT2_ACTTE Tenebrosin C - Actinia tenebrosa (Australian red waratah sea anemone).
2.	sp|P15310|PLC1_CLOPE Phospholipase C precursor (EC 3.1.4.3) (PLC) (Phosphatidylcholine cholinephosphohydrolase) (Alpha-toxin) (Hemolysin) (Lecithinase) - Clostridium perfringens.

Figure 7: Results for homology search: The initial results list the query sequences (if more than one query has been submitted) which are links which take the user to the results page with the list of actual homologs.

- The list of homologs in a page-wise format, iteration by iteration, is displayed as shown below in Figure 8.


http://sri:8080/TkbProject/servlet/PrevNextServlet?value=next&iter=1&def=sp|P15310|PLC1_CLOPE P - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://sri:8080/TkbProject/servlet/PrevNextServlet?value=next&iter=1&def=sp|P15310|PLC1_CLOPE%20Phospholipase%20C%20precursor%20(EC%203.1.4.3)%20(PL
Go Links
WinAgent Record Play E-mail Pop-ups
Done
Local intranet
start
Local Disk (C:)
Command Pro...
Tomcat
Untitled Docu...
http://sri:808...
untitled - Paint
10:44 AM

HOMOLOGY SEARCH RESULTS

Program: blastp 2.2.9 [May-01-2004]
Query: sp|P15310|PLC1_CLOPE Phospholipase C precursor (EC 3.1.4.3) (PLC) (Phosphatidylcholine cholinephosphohydrolase) (Alpha-toxin) (Hemolysin) (Lecithinase) - Clostridium perfringens.
Length: 398
Database: C:\tkb_db\tkb.txt

ITERATION: 2

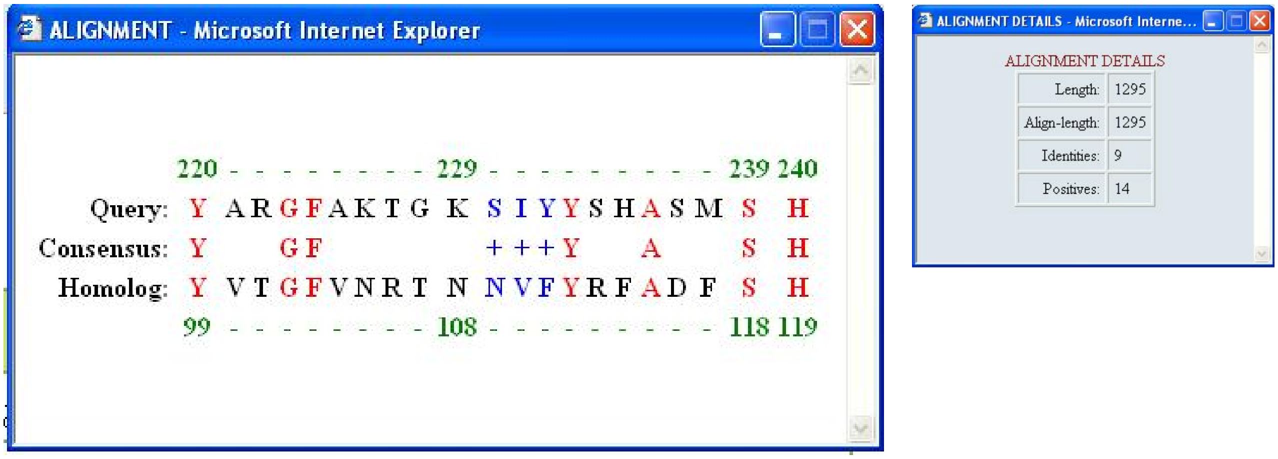
<<Prev Iteration

HIT	ID	DEFINITION	SCORE (bits)	E-VALUE	ALIGNMENT
1	gnl|BL_ORD_ID|2	sp|P15310|PLC1_CLOPE Phospholipase C precursor (EC 3.1.4.3) (PLC) (Phosphatidylcholine cholinephosphohydrolase) (Alpha-toxin) (Hemolysin) (Lecithinase) - Clostridium perfringens.	843.988	0	View Alignment View Details
2	gnl|BL_ORD_ID|5	sp|P10845|BXA1_CLOBO Botulinum neurotoxin type A precursor (EC 3.4.24.69) (BoNT/A) (Bontoxilysin A) (BOTOX) [Contains: Botulinum neurotoxin A, light- chain; Botulinum neurotoxin A, heavy-chain] - Clostridium botulinum. new - FASTA	21.5871	4.92847	View Alignment View Details
3	gnl|BL_ORD_ID|37	sp|P10149|SLTA_BPH30 Shiga toxin A-chain precursor (EC 3.2.2.22) (rRNA N-glycosidase) - Bacteriophage H30.	21.2019	5.49312	View Alignment View Details
4	gnl|BL_ORD_ID|38	sp|P08026|SLTA_BPH19 Shiga-like toxin I subunit A precursor (EC 3.2.2.22) (Verotoxin 1 subunit A) (SLT-I) (rRNA N-glycosidase) - Bacteriophage H19B.	21.2019	5.49312	View Alignment View Details

<<Prev Iteration

Figure 8: Results for Homology Search: The results of the each query are reproduced in a easy-to-use tabular format that shows all the required information, along with the new homologs identified in subsequent iterations, that are highlighted as shown here.

- Each homolog has the following information:
    1. A link to the NCBI/Swiss-Prot entry depending on whether the homologs are from "nr" or "TKB" respectively.
    2. New homologs in subsequent iterations are identified.
    3. Each homolog has the following information:
        - Score for each homolog
        - E-value for each homolog
        - Pair-wise alignment of the query and homolog sequence, showing the positives, identities and gaps. (shown in Figure 9 below)
        - Alignment details (shown in the Figure 9 below)


ALIGNMENT - Microsoft Internet Explorer
ALIGNMENT DETAILS - Microsoft Interne...

```
            220 - - - - - - - - 229 - - - - - - - - - 239 240
     Query: Y A R G F A K T G K S I Y Y S H A S M S H
 Consensus: Y     G F           + + + Y     A     S H
   Homolog: Y V T G F V N R T N N V F Y R F A D F S H
            99 - - - - - - - - 108 - - - - - - - - - 118 119
```

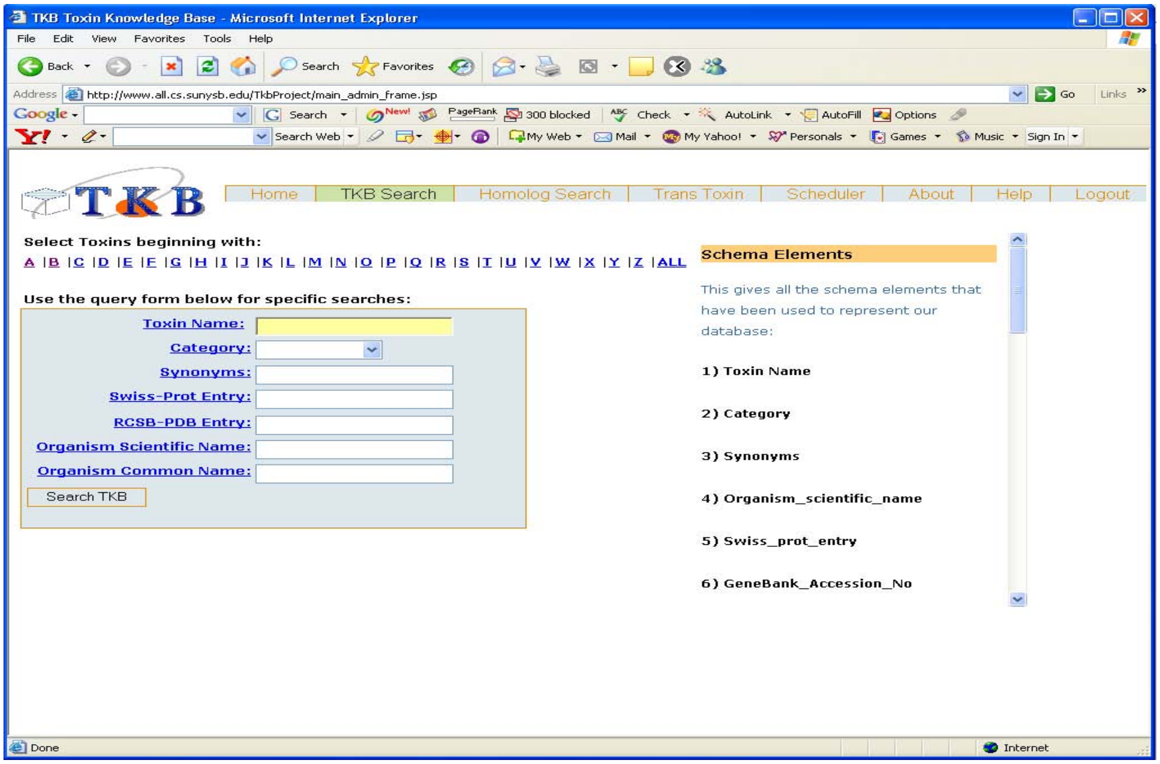
ALIGNMENT DETAILS

Length:	1295
Align-length:	1295
Identities:	9
Positives:	14

Figure 9: Pair-wise alignment and Details: The pair-wise alignment shows the identities(the residues marked red) and the positives (the residues marked blue). The alignment details are also provided.

TKB Search

- The TKB Search interface is useful to view the toxin data stored in the TKB.
- The user can browse through the toxins alphabetically. The user can also search for particular toxins by specifying certain filter criteria as shown in the Figure 10 below.



Address http://www.all.cs.sunysb.edu/TkbProject/main_admin_frame.jsp

Select Toxins beginning with:
A |B |C |D |E |F |G |H |I |J |K |L |M |N |O |P |Q |R |S |T |U |V |W |X |Y |Z |ALL

Use the query form below for specific searches:

Toxin Name:
Category:
Synonyms:
Swiss-Prot Entry:
RCSB-PDB Entry:
Organism Scientific Name:
Organism Common Name:
Search TKB

Schema Elements

This gives all the schema elements that have been used to represent our database:

1) Toxin Name

2) Category

3) Synonyms

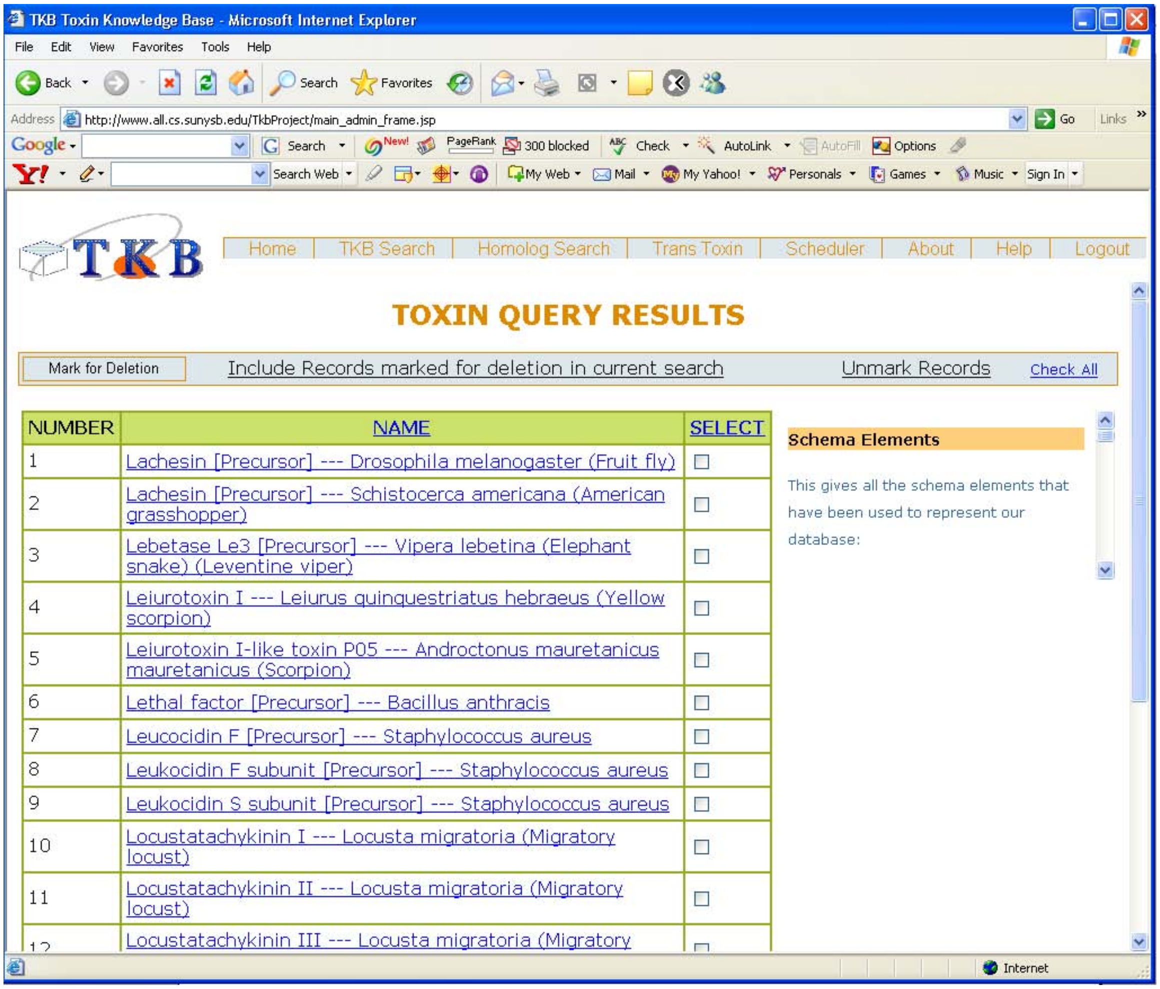
4) Organism_scientific_name

5) Swiss_prot_entry

6) GeneBank_Accession_No

Figure 10: TKB Search Interface: The above figure displays the search options

The search results containing the names of the toxins satisfying the chosen filter criteria are displayed as shown in the following figures (Figures 11).


TKB Toxin Knowledge Base - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://www.all.cs.sunysb.edu/TkbProject/main_admin_frame.jsp
Go Links
Google Search PageRank 300 blocked Check AutoLink AutoFill Options
Search Web My Web Mail My Yahoo! Personals Games Music Sign In
TKB
Home TKB Search Homolog Search Trans Toxin Scheduler About Help Logout
Internet

TOXIN QUERY RESULTS

Mark for Deletion | Include Records marked for deletion in current search | Unmark Records | Check All

NUMBER	NAME	SELECT
1	Lachesin [Precursor] --- Drosophila melanogaster (Fruit fly)	□
2	Lachesin [Precursor] --- Schistocerca americana (American grasshopper)	□
3	Lebetase Le3 [Precursor] --- Vipera lebetina (Elephant snake) (Leventine viper)	□
4	Leiurotoxin I --- Leiurus quinquestriatus hebraeus (Yellow scorpion)	□
5	Leiurotoxin I-like toxin P05 --- Androctonus mauretanicus mauretanicus (Scorpion)	□
6	Lethal factor [Precursor] --- Bacillus anthracis	□
7	Leucocidin F [Precursor] --- Staphylococcus aureus	□
8	Leukocidin F subunit [Precursor] --- Staphylococcus aureus	□
9	Leukocidin S subunit [Precursor] --- Staphylococcus aureus	□
10	Locustatachykinin I --- Locusta migratoria (Migratory locust)	□
11	Locustatachykinin II --- Locusta migratoria (Migratory locust)	□
12	Locustatachykinin III --- Locusta migratoria (Migratory	□

Schema Elements

This gives all the schema elements that have been used to represent our database:

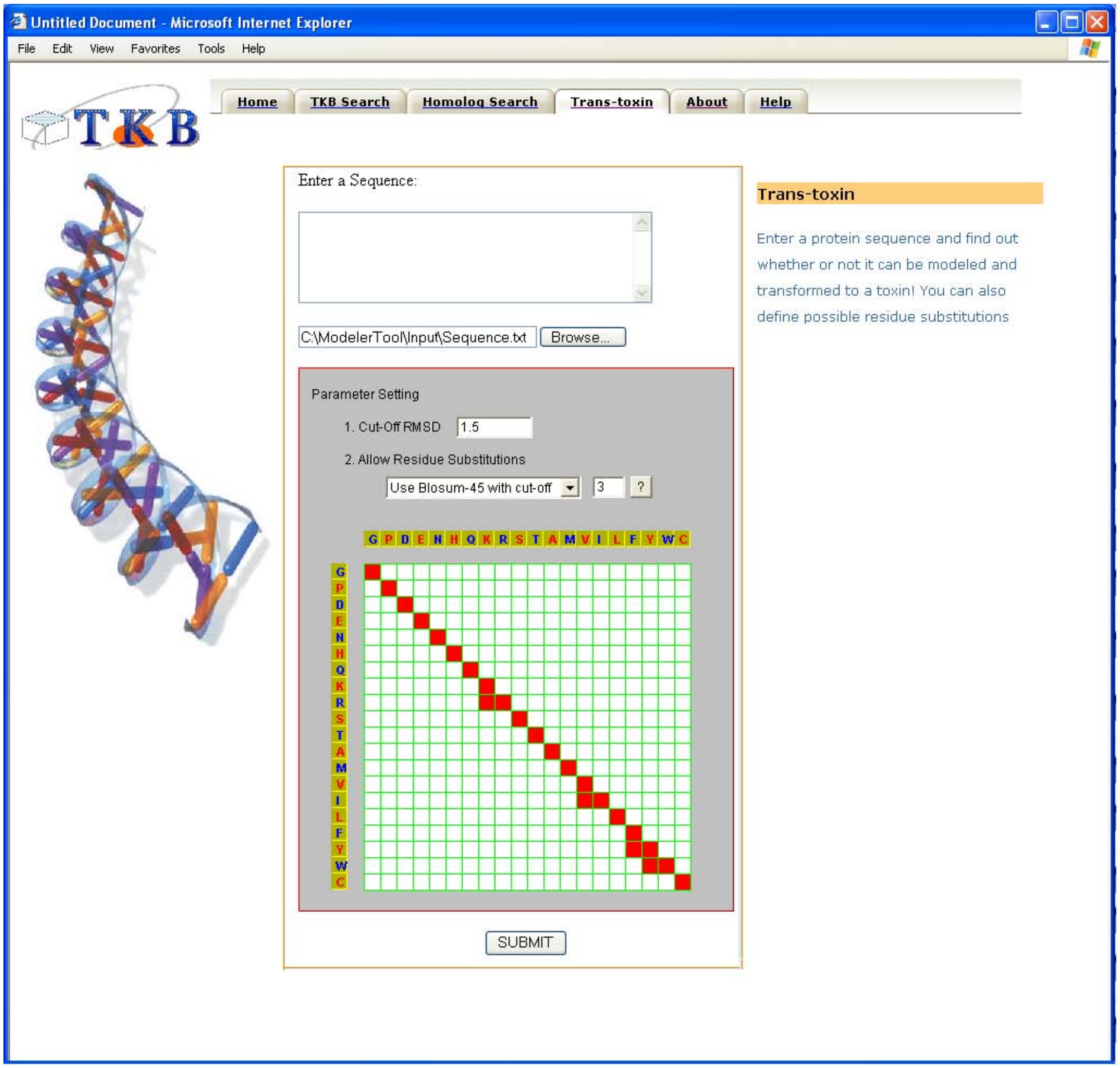
Figure 11: TKB Search Results: The above are the results that are displayed when the user searched for toxins starting with alphabet 'L'

- The details of the toxin can be seen by clicking on that particular toxin name in the search results. This was illustrated in the previous section on the system design (figure 3).

Trans-Toxin (Mu-Toxin)

- This interface lets the user investigate whether a protein can be transformed into a toxin.

- It accepts the protein sequence from the user, either as file or a text string. The user interface is shown in Figure 12.



Trans-toxin

Enter a protein sequence and find out whether or not it can be modeled and transformed to a toxin! You can also define possible residue substitutions

Enter a Sequence:

C:\ModelerTool\Input\Sequence.txt

Parameter Setting

1. Cut-Off RMSD 1.5

2. Allow Residue Substitutions

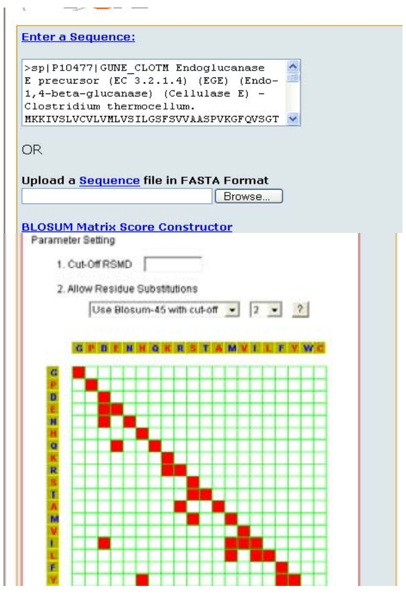
Use Blosum-45 with cut-off 3

	G	P	D	E	N	H	Q	K	R	S	T	A	M	V	I	L	F	Y	W	C
G	🟥																			
P		🟥																		
D			🟥																	
E				🟥																
N					🟥															
H						🟥														
Q							🟥													
K								🟥												
R								🟥	🟥											
S										🟥										
T											🟥									
A												🟥								
M													🟥							
V														🟥						
I														🟥	🟥					
L																🟥				
F																	🟥			
Y																	🟥	🟥		
W																		🟥	🟥	
C																				🟥

Figure 12: MuToxin (Trans-toxin) Interface.

A sample output from the Mutoxin (trans-toxin) interface is shown in the figure below. It provides two results. One is a tabulation of all the possible matches of the input protein against the templates of active sites in the knowledge base with the RMSD values, BLOSUM scores and model output by Modeller to get these results. It also provides a detailed view (highlighted in the picture), where in the user can see the matching active site residues, and the corresponding BLOSUM scores (if they were selected by the user;

otherwise the customized score value is calculated from the input as highlighted by the user).



Enter a Sequence:

```
>sp|P10477|GUNE_CLOTM Endoglucanase
E precursor (EC 3.2.1.4) (EGE) (Endo-
1,4-beta-glucanase) (Cellulase E) -
Clostridium thermocellum.
MKKIVSLVCVLVMLVSILGSFSVVAASPVKGFQVSGT
```

OR

Upload a Sequence file in FASTA Format

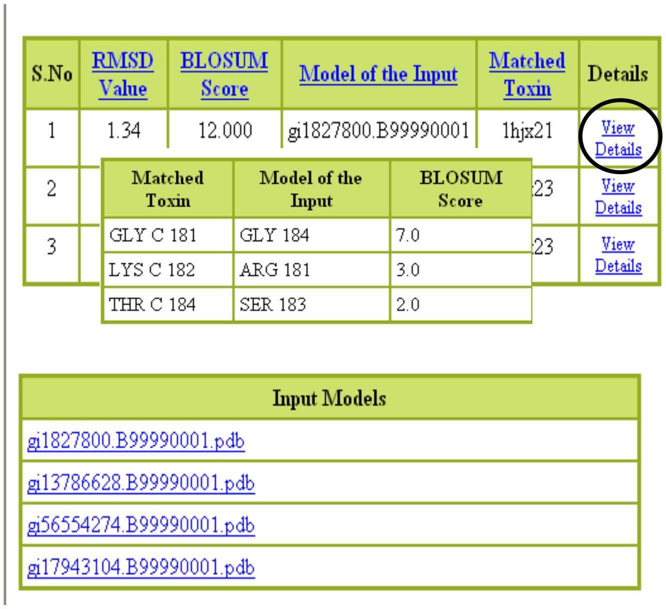
BLOSUM Matrix Score Constructor

Parameter Setting

1. Cut-Off RSMD

2. Allow Residue Substitutions: Use Blosum-45 with cut-off 2

Browse...
?
G P D E N H Q K R S T A M V I L F Y W C
G P D E N H Q K R S T A M V I L F V



S.No	RMSD Value	BLOSUM Score	Model of the Input	Matched Toxin	Details
1	1.34	12.000	gi1827800.B99990001	1hjx21	View Details
2				23	View Details
3				23	View Details

Matched Toxin	Model of the Input	BLOSUM Score
GLY C 181	GLY 184	7.0
LYS C 182	ARG 181	3.0
THR C 184	SER 183	2.0

Input Models
gi1827800.B99990001.pdb
gi13786628.B99990001.pdb
gi56554274.B99990001.pdb
gi17943104.B99990001.pdb

Administrative interfaces

The update interface lets the Administrator maintain a consistent and up-to-date knowledge base. The updates to the toxin knowledge base can be done in two ways:

- Automated: This includes the following two tasks:
    - Update TKB
    - Update NR
- User-initiated

These tasks are explained in detail in the following sections.

Update TKB

The TKB has to be updated whenever new proteins are added to the NR database. New entries in the NR database could mean additional homologs for the toxins in the TKB. An update on the TKB is performed by blasting each toxin against the most recent additions to the NR database.

Given a sequence from the TKB, the following steps in this task have been automated.

i. Blast the Query sequence against the most recently available updates to the NR database.
ii. Process the list of homologs to obtain the list of homologs that are relevant to the input to be given to the modeler. This processing involves filtering based on e-values and identity cut-offs.
iii. Store the desired set of homologs in the TKB.

Update NR

A copy of the NR database is being maintained. Sequences are added to the NR database whenever new proteins are released. This addition of sequences involves matching new sequences to existing ones and appending and/or inserting new entries in the NR. Throughout the process, caution is taken to maintain the non-redundant property of the NR database.

User initiated update of TKB

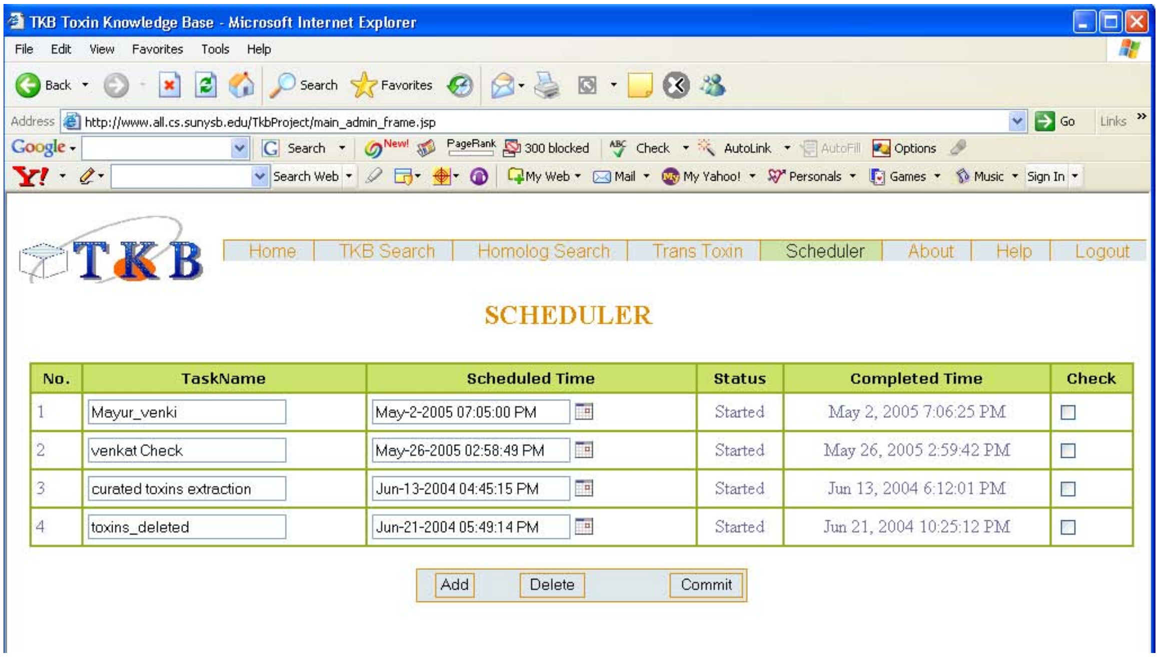
This is the step when a new toxin has been identified and hence an entry is made in the TKB along with the following information:

- PDB ID: Using WinAgent, Swiss-Prot is searched with the toxin's accession number (for example, P10844 for Botulinum neurotoxin type B), and then the PDB ID's for the toxin are extracted.
- Active site information: Using WinAgent, PDB, PDBSUM, and LPC databases are searched using the PDB ID of a toxin, and then the active site information, if there is any is extracted.
- Models: For each homolog, the toxin to which it is homologous is known. If the toxin structure information is available, a model is built for the homolog based on the alignment of the toxin and the homolog as well as the structure of the toxin using MODELLER. MODELLER is a well-known comparative modeling tool. It has its own script language to control the modeling process. Using a program to generate the script, the modeling process is automated.

Scheduler

- The above-mentioned update tasks are long-running tasks, to be performed on a periodic basis. A Scheduler application has been developed to handle the updates.
- The Administrator is provided with a facility to schedule these tasks to be executed at a specified time and interval. It is the responsibility of the Scheduler application to execute the tasks thereafter, at the predefined time and interval on a regular basis.

The user interface for the scheduler is shown in Figure 13 below:



SCHEDULER

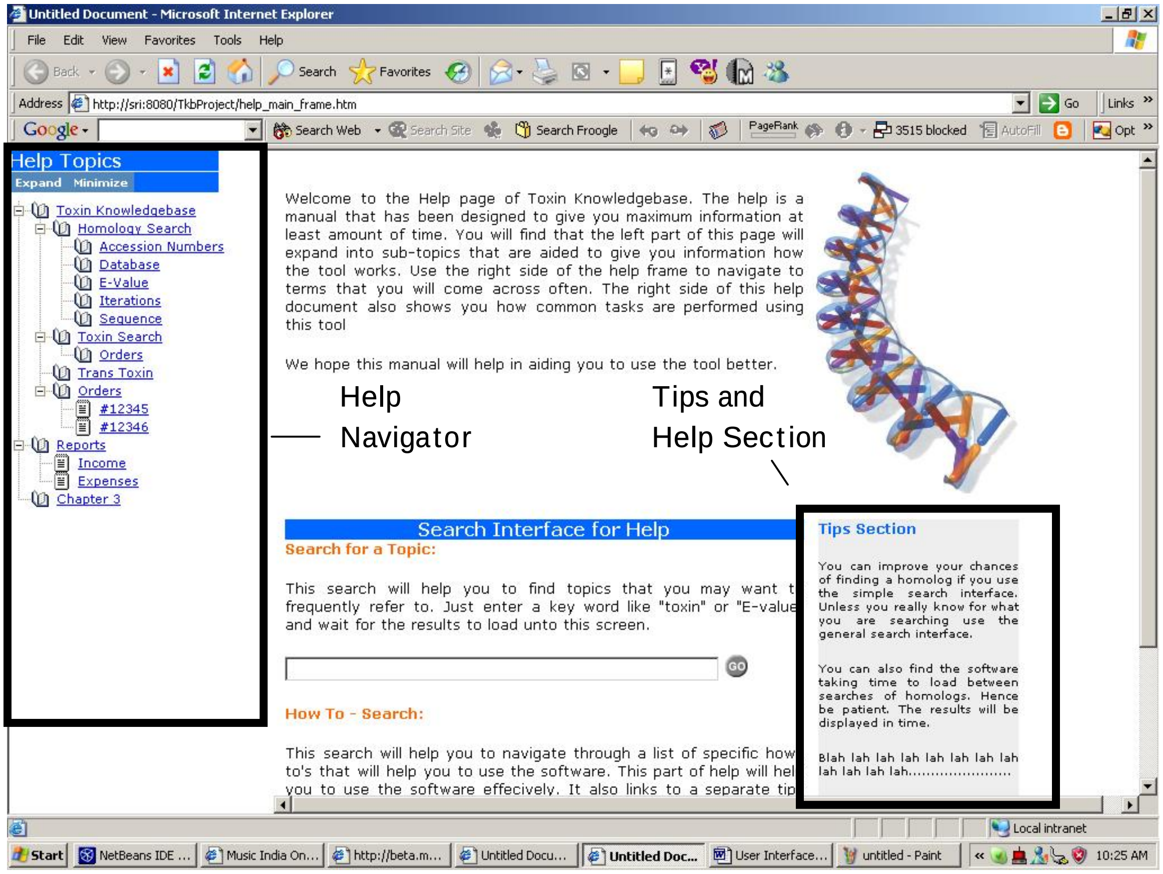
No.	TaskName	Scheduled Time	Status	Completed Time	Check
1	Mayur_venki	May-2-2005 07:05:00 PM	Started	May 2, 2005 7:06:25 PM	□
2	venkat Check	May-26-2005 02:58:49 PM	Started	May 26, 2005 2:59:42 PM	□
3	curated toxins extraction	Jun-13-2004 04:45:15 PM	Started	Jun 13, 2004 6:12:01 PM	□
4	toxins_deleted	Jun-21-2004 05:49:14 PM	Started	Jun 21, 2004 10:25:12 PM	□

Figure 13: The Update Interface: The administrator can schedule the Update Task by just selecting the Task from a menu. The administrator decides the start date for Scheduling an event.

- The information about the status of the execution of these scheduled tasks is stored in the database. The Administrator will be provided with an interface to see the status of these scheduled tasks and also update the task information. This will include the time at which the task is to be executed or the interval between two successive executions of the task. The user will also be able to delete a pre-defined task.

Help interface

Help pages are provided for each module of the interface as shown in the screen shot (Figure 14). The help consists of hierarchical tree structure to help users to navigate and also a separate section on help for all information regarding the project and the site. Apart from this an interesting and user-friendly feature of the help pages is the ability to display information “inline” – especially when forms are being used to gather user-typed input. This improves the functionality as well as interactivity of the user.


Untitled Document - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address http://sri:8080/TkbProject/help_main_frame.htm
Go Links
Google Search Web Search Site Search Froogle PageRank 3515 blocked AutoFill Opt
Help Topics
Expand Minimize
Toxin Knowledgebase
Homology Search
Accession Numbers
Database
E-Value
Iterations
Sequence
Toxin Search
Orders
Trans Toxin
Orders
#12345
#12346
Reports
Income
Expenses
Chapter 3
Help Navigator
Tips and Help Section
GO
Local intranet
Start NetBeans IDE ... Music India On... http://beta.m... Untitled Docu... Untitled Doc... User Interface... untitled - Paint
10:25 AM

Welcome to the Help page of Toxin Knowledgebase. The help is a manual that has been designed to give you maximum information at least amount of time. You will find that the left part of this page will expand into sub-topics that are aided to give you information how the tool works. Use the right side of the help frame to navigate to terms that you will come across often. The right side of this help document also shows you how common tasks are performed using this tool

We hope this manual will help in aiding you to use the tool better.

Search Interface for Help

Search for a Topic:

This search will help you to find topics that you may want t frequently refer to. Just enter a key word like "toxin" or "E-value and wait for the results to load unto this screen.

How To - Search:

This search will help you to navigate through a list of specific how to's that will help you to use the software. This part of help will hel you to use the software effecively. It also links to a separate tip

Tips Section

You can improve your chances of finding a homolog if you use the simple search interface. Unless you really know for what you are searching use the general search interface.

You can also find the software taking time to load between searches of homologs. Hence be patient. The results will be displayed in time.

Blah lah lah lah lah lah lah lah lah lah lah lah.......................

Figure 14: Help Interface Design: The help interface is designed to be user friendly having a tree that helps the user to navigate easily through out the help. The help interface also gives useful tips and strategies to use the site as a whole.

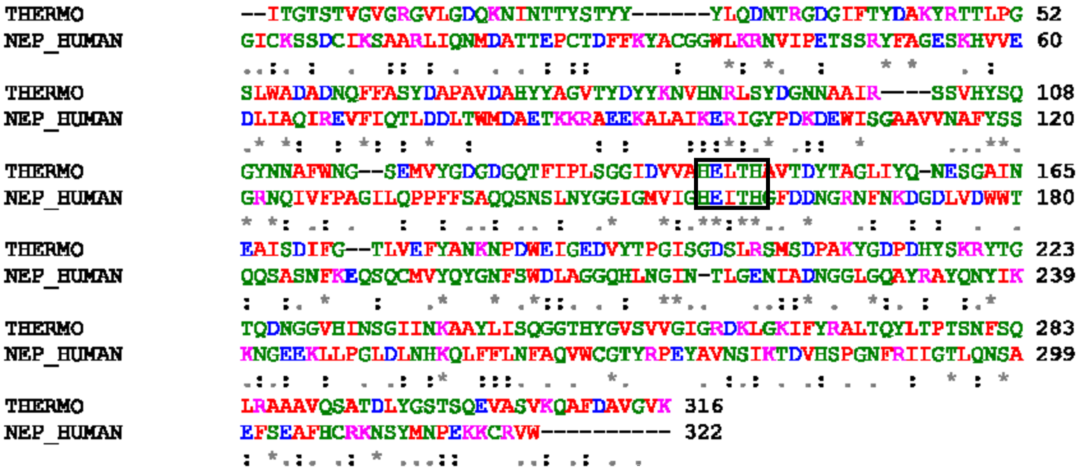
Case Studies

This section provides briefly some observed results validated by biochemical experiments, as well as some results that do not yet have a thorough biochemical validation. First we present the result of comparing Thermolysin, Neprilysin and Botulinum neurotoxin type E, of which Thermolysin and Neprilysin are non neurotoxic proteins, whereas Botulinum neurotoxin type E is a potent neurotoxin. Next we present the results of comparing Endoglucanase, with Chitinase of which Chitinase is a known toxin. For the former set of proteins, several studies have focused on the similarities of all the three proteins; however, more focus has been laid on the similarity of Thermolysin and Neprilysin which are known zinc binding proteases. The latter set, although a couple of studies have been published, there has been no conclusive evidence through biochemical experiments that these two proteins are indeed similar.

Similarity of the Reaction Mechanism in Thermolysin, Neprilysin, and Botulinum neurotoxin Type E

Thermolysin, Neprilysin and Botulinum share a same motif HEXXH + E and it is speculated that they have similar reaction mechanism. The functional similarity of Thermolysin and Neprilysin has long been recognized due to a relatively significant sequence homology between the two proteins, as is shown in Figure 15-a. The statistics of the alignment is shown in the following table.

Length	333
Number of identical matches	37
Number of positive matches	89



```
THERMO          --ITGTSTVGVGRGVLGDQKNINTTYSTYY------YLQDNTRGDGIFTYDAKYRTTLPG 52
NEP_HUMAN       GICKSSDCIKSAARLIQNMDATTEPCTDFFKYACGGWLKRNVIPETSSRYFAGESKHVVE 60
                ..:. :  .  :: : .  . . : ::      :   *: *.  :    * *   . :

THERMO          SLWADADNQFFASYDAPAVDAHYYAGVTYDYYKNVHNRLSYDGNNAAIR----SSVHYSQ 108
NEP_HUMAN       DLIAQIREVFIQTLDDLTWMDAETKKRAEEKALAIKERIGYPDKDEWISGAAVVNAFYSS 120
                .* *:  : *: : *  :         : :    :::*:.* .::  *      ...**.

THERMO          GYNNAFWNG--SEMVYGDGDGQTFIPLSGGIDVVAHELTHAVTDYTAGLIYQ-NESGAIN 165
NEP_HUMAN       GRNQIVFPAGILQPPFFSAQQSNSLNYGGIGMVIGHELTHGFDDNGRNFNKDGDLVDWWT 180
                * *: .: .   :  : ..: .. :  .*   *:.**:**.. *   .:  : :  .  .

THERMO          EAISDIFG--TLVEFYANKNPDWEIGEDVYTPGISGDSLRSMSDPAKYGDPDHYSKRYTG 223
NEP_HUMAN       QQSASNFKEQSQCMVYQYGNFSWDLAGGQHLNGIN-TLGENIADNGGLGQAYRAYQNYIK 239
                :  :. *   :   .*   * .*::. . :  **..   ..::* .  *:. :  :.*

THERMO          TQDNGGVHINSGIINKAAYLISQGGTHYGVSVVGIGRDKLGKIFYRALTQYLTPTSNFSQ 283
NEP_HUMAN       KNGEEKLLPGLDLNHKQLFFLNFAQVWCGTYRPEYAVNSIKTDVHSPGNFRIIGTLQNSA 299
                .:.:  :  . .: :*  :::. . .  *.     . :.: . .: . .  :  * : *

THERMO          LRAAAVQSATDLYGSTSQEVASVKQAFDAVGVK 316
NEP_HUMAN       EFSEAFHCRKNSYMNPEKKCRVW---------- 322
                : *.:. .: * ...::    ..: .: . .
```

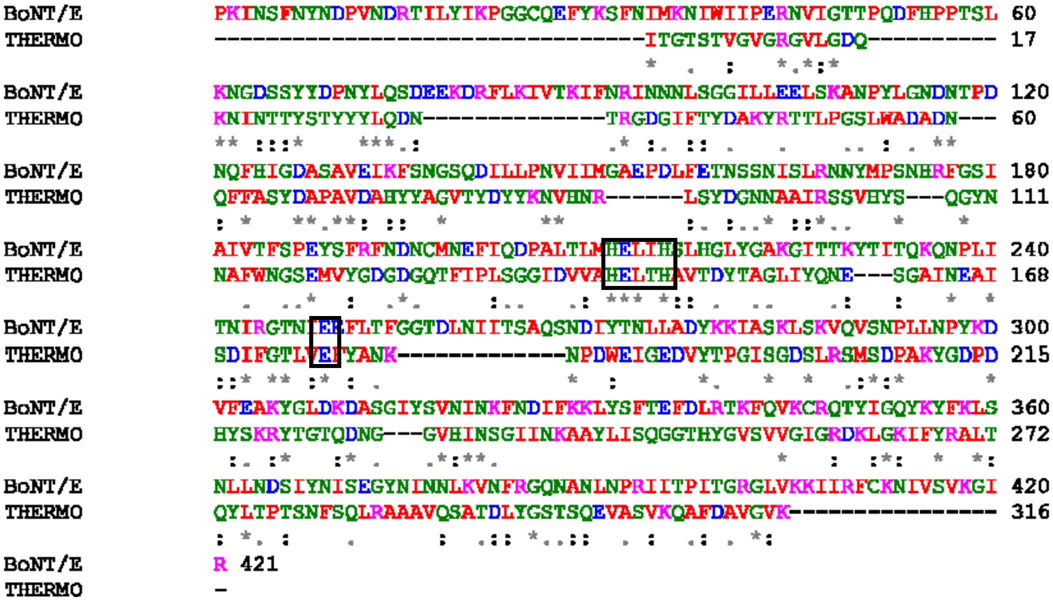
Figure 15-a. Homology between Thermolysin and Neprilysin

In contrast, the sequence homology between Thermolysin, Neprilysin and Botulinum is low, which can be seen from the alignment between Thermolysin and Botulinum neurotoxin serotype E in Figure 15-b. The statistics of the alignment is listed below.

Length	421
Number of identical matches	56

Number of positive matches	88

Because the alignment between Thermolysin and Botulinum does not indicate that they are closely related, one may conclude that they may not share any significant structural or functional similarity. Structural alignment of the full structures of Thermolysin, Neprilysin and Botulinum also fails to reveal the functional similarity between them. By concentrating on the active sites, the Trans-toxin (Mutoxin interface) is able to find that the active sites of Thermolysin and Neprilysin are similar to that of Botulinum, as shown in Figure 15-c, and therefore predicts that non-toxic proteins Thermolysin and Neprilysin might be mutated to function like Botulinum neurotoxin.



```
BoNT/E          PKINSFNYNDPVNDRTILYIKPGGCQEFYKSFNIMKNIWIIPERNVIGTTPQDFHPPTSL 60
THERMO          ---------------------------------ITGTSTVGVGRGVLGDQ---------- 17
                                                 *  .  :   *.*:*

BoNT/E          KNGDSSYYDPNYLQSDEEKDRFLKIVTKIFNRINNNLSGGILLEELSKANPYLGNDNTPD 120
THERMO          KNINTTYSTYYYLQDN--------------TRGDGIFTYDAKYRTTLPGSLWADADN--- 60
                ** :::*    ***.:              .* :. :: .   .    .. : . **

BoNT/E          NQFHIGDASAVEIKFSNGSQDILLPNVIIMGAEPDLFETNSSNISLRNNYMPSNHRFGSI 180
THERMO          QFFASYDAPAVDAHYYAGVTYDYYKNVHNR------LSYDGNNAAIRSSVHYS---QGYN 111
                : *   **.**: ::  *       **         :. :..* ::*..   *    *

BoNT/E          AIVTFSPEYSFRFNDNCMNEFIQDPALTLMHELTHSLHGLYGAKGITTKYTITQKQNPLI 240
THERMO          NAFWNGSEMVYGDGDGQTFIPLSGGIDVVAHELTHAVTDYTAGLIYQNE---SGAINEAI 168
                  .  ..*  :  .*.     :..   .: *** *:: .  ..    .:   :   *  *

BoNT/E          TNIRGTNIEEFLTFGGTDLNIITSAQSNDIYTNLLADYKKIASKLSKVQVSNPLLNPYKD 300
THERMO          SDIFGTLVEFYANK-------------NPDWEIGEDVYTPGISGDSLRSMSDPAKYGDPD 215
                ::* ** :* : .              *  :      *.   *  *  .:*:*      *

BoNT/E          VFEAKYGLDKDASGIYSVNINKFNDIFKKLYSFTEFDLRTKFQVKCRQTYIGQYKYFKLS 360
THERMO          HYSKRYTGTQDNG---GVHINSGIINKAAYLISQGGTHYGVSVVGIGRDKLGKIFYRALT 272
                 :. :*   :* .   .*:**.                     *   :  :*:  *  *:

BoNT/E          NLLNDSIYNISEGYNINNLKVNFRGQNANLNPRIITPITGRGLVKKIIRFCKNIVSVKGI 420
THERMO          QYLTPTSNFSQLRAAAVQSATDLYGSTSQEVASVKQAFDAVGVK---------------- 316
                : *. :    .      :  .:: *..::  . :  .: . *:

BoNT/E          R 421
THERMO          -
```

Figure 15-b. Remote homology between Thermolysin and Botulinum E

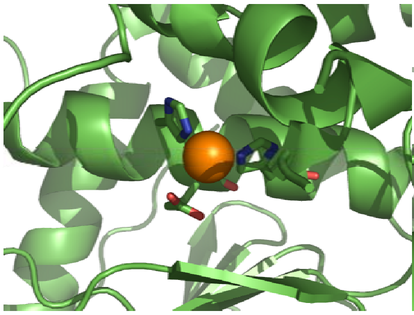


Figure 15-c(1): Active site of Botulinum neurotoxin E light chain (PDB:1T3A), residues shown H211, E212 and H215

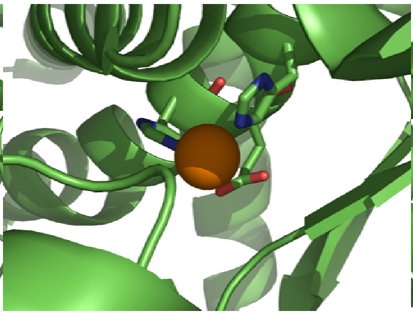


Figure 15-c(2): Active site of human Neprilysin (PDB:1DMT), residues shown H583, E584 and H587

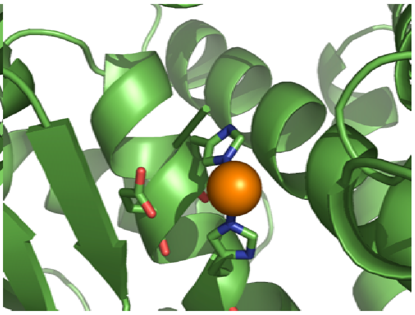


Figure15-c(3): Active site of Thermolysin (PDB:1TLP), residues shown H142, E143 and H146

Similar Reactive Mechanism of Endoglucanase and Chitinase

We chose the sequence of Endoglucanase since it is an important protein that binds to cellulose, as well as having a multidomain enzymatic characteristic. A sequence comparison between Endoglucanase (Swissprot ID: P10477) and Chitinase (PDB ID: 1HJX) does not give a good hint about their similarity, as is shown in Figure 15-d. The length of the alignment, the number of identical matches and the number of positive matches are as follows.

Length	696
Number of identical matches	74
Number of positive matches	94

Moreover, because no structure is available for Endoglucanase, one may stop at the sequence level and conclude that Endoglucanase do not share functional similarity with Chitinase without further structural analysis.

Trans-toxin tries to build a model of Endoglucanase based on its sequence homology with proteins whose structures have been determined using Modeller. Then at the structure level, a putative active site in Endoglucanase is similar to that of Chitinase, as shown in Figure 15-e thus reveals that Endoglucanase is potentially a candidate to be transformed to Chitinase.

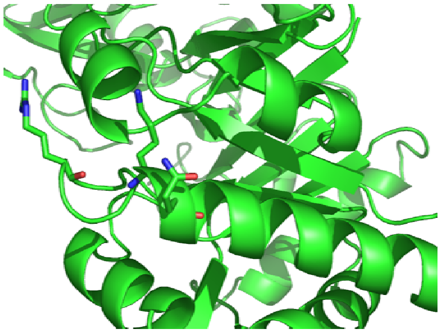


Figure 15-e (1): Active site of Chitinase (PDB: 1HJX), residues shown are R144, K147 and Q148

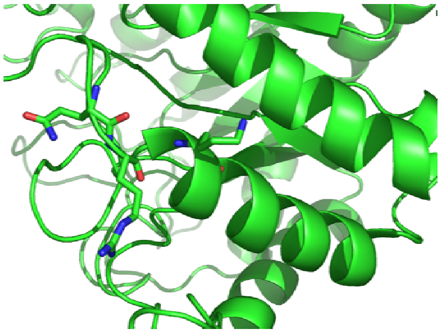
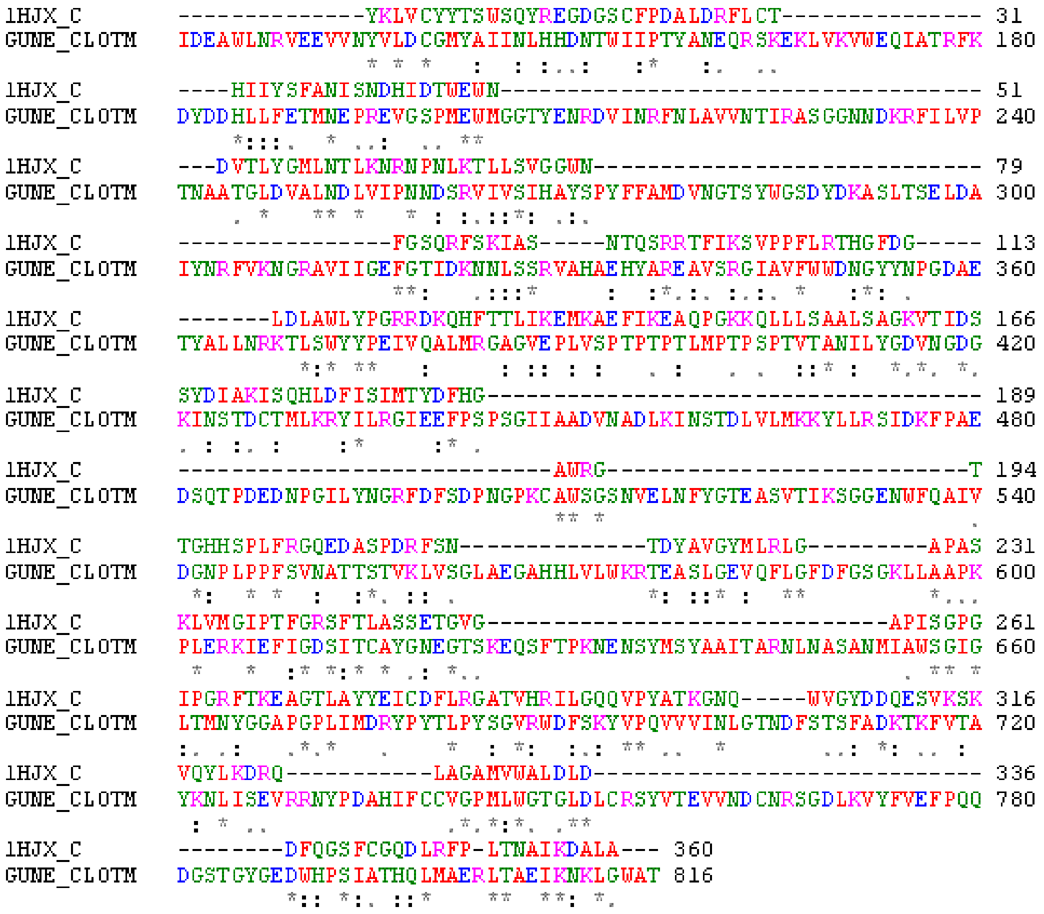


Figure 15-e (2): Putative active site of Endoglucanase E precursor (Swissprot: P10477), residues shown are R118, 20, and Q117.



```
1HJX_C          --------------YKLVCYYTSWSQYREGDGSCFPDALDRFLCT--------------- 31
GUNE_CLOTM      IDEAWLNRVEEVVNYVLDCGMYAIINLHHDNTWIIPTYANEQRSKEKLVKVWEQIATRFK 180
                              * * *   :  : :..:   :*   :.  ..
1HJX_C          ----HIIYSFANISNDHIDTWEWN------------------------------------ 51
GUNE_CLOTM      DYDDHLLFETMNEPREVGSPMEWMGGTYENRDVINRFNLAVVNTIRASGGNNDKRFILVP 240
                    *:::.  * ..:  ..  **
1HJX_C          ---DVTLYGMLNTLKNRNPNLKTLLSVGGWN----------------------------- 79
GUNE_CLOTM      TNAATGLDVALNDLVIPNNDSRVIVSIHAYSPYFFAMDVNGTSYWGSDYDKASLTSELDA 300
                    . *   ** *   * : :.::*: .:.
1HJX_C          ----------------FGSQRFSKIAS-----NTQSRRTFIKSVPPFLRTHGFDG----- 113
GUNE_CLOTM      IYNRFVKNGRAVIIGEFGTIDKNNLSSRVAHAEHYAREAVSRGIAVFWWDNGYYNPGDAE 360
                                **:   .:::*     :  :*.:. :.:. *   :*: .
1HJX_C          -------LDLAWLYPGRRDKQHFTTLIKEMKAEFIKEAQPGKKQLLLSAALSAGKVTIDS 166
GUNE_CLOTM      TYALLNRKTLSWYYPEIVQALMRGAGVEPLVSPTPTPTLMPTPSPTVTANILYGDVNGDG 420
                         *:* **   :     : :: : :   . :   . .  ::* :  *.*. *.
1HJX_C          SYDIAKISQHLDFISIMTYDFHG------------------------------------- 189
GUNE_CLOTM      KINSTDCTMLKRYILRGIEEFPSPSGIIAADVNADLKINSTDLVLMKKYLLRSIDKFPAE 480
                . : :. :    :*     :* .
1HJX_C          ----------------------------AWRG---------------------------T 194
GUNE_CLOTM      DSQTPDEDNPGILYNGRFDFSDPNGPKCAWSGSNVELNFYGTEASVTIKSGGENWFQAIV 540
                                            ** *                           .
1HJX_C          TGHHSPLFRGQEDASPDRFSN--------------TDYAVGYMLRLG---------APAS 231
GUNE_CLOTM      DGNPLPPFSVNATTSTVKLVSGLAEGAHHLVLWKRTEASLGEVQFLGFDFGSGKLLAAPK 600
                 *:  * *  :  :*. :: .              *: ::* :  **         *...
1HJX_C          KLVMGIPTFGRSFTLASSETGVG------------------------------APISGPG 261
GUNE_CLOTM      PLERKIEFIGDSITCAYGNEGTSKEQSFTPKNENSYMSYAAITARNLNASANMIAWSGIG 660
                 *   *  :* *:* * .: *..                               . ** *
1HJX_C          IPGRFTKEAGTLAYYEICDFLRGATVHRILGQQVPYATKGNQ-----WVGYDDQESVKSK 316
GUNE_CLOTM      LTMNYGGAPGPLIMDRYPYTLPYSGVRWDFSKYVPQVVVINLGTNDFSTSFADKTKFVTA 720
                :. .:   .*.*   .    *  : *:  :.: ** ..  *       ..: *: .. :
1HJX_C          VQYLKDRQ-----------LAGAMVWALDLD----------------------------- 336
GUNE_CLOTM      YKNLISEVRRNYPDAHIFCCVGPMLWGTGLDLCRSYVTEVVNDCNRSGDLKVYFVEFPQQ 780
                 : * ..             .*.*:*. .**
1HJX_C          --------DFQGSFCGQDLRFP-LTNAIKDALA--- 360
GUNE_CLOTM      DGSTGYGEDWHPSIATHQLMAERLTAEIKNKLGWAT 816
                        *:: *:. ::*    **  **: *.
```

Figure 15-d. Remote homology between Endoglucanase and Chitinase

Profiling active sites in proteins

Active sites in proteins are three dimensional substructures that cause them to perform their function. In TKB, finding substructures in a protein that are ``similar'' to the active sites of some protein is the key step to decide whether the protein can be transformed into the toxin. Active sites can be grouped into families whose members are related by similarity of their functions. Since similar sites exhibit variability in their physico-chemical and structural features, statistical profiling methods capture the shared features robustly in the presence of such variations. Such methods can find substructures that possess the features shared by all family members but not those varying from member to member, which might otherwise be missed by comparison to individual active sites.

We studied the possibility of adapting Profile Hidden Markov Models (PHMMs) that have been successfully used for analyzing biological sequences, to statistically profile active site families. Since PHMMs can only profile one-dimensional sequences, we developed a serialization of the three dimensional active sites that capture certain shared physico-chemical and geometric features of the family. PHMM parameters are learnt using these serialized sequences. While traditional PHMM learning algorithms deal with discrete physico-chemical feature only, we expanded it to include geometric features drawn from a continuous probability distribution.

Profiling protein families from partially aligned sequences

Profile-based homology search methods can detect more remote homologues than pair-wise based methods such as BLAST. Among all profile-base methods, Profile Hidden Markov Models (PHMMs) is accepted as a powerful technique. Extant PHMM training approaches either use completely unaligned or aligned sequences. The PHMMs resulting from these two training approaches present contrasting tradeoffs w.r.t. alignment information and the accuracy of the search outcome.

We developed a PHMM based technique for modeling protein families from partially aligned sequences, for which alignment information is available for subsequences of the training protein sequences. By exploiting the observation that partially aligned sequences give rise to independent subsequences, PHMMs corresponding to these subsequences are composed to build PHMMs for the entire sequences. An interesting aspect of the technique is that it gives rise to a family of PHMMs which are parameterized w.r.t. the alignment information, spanning the range from PHMMs trained from unaligned sequences at one extreme and those from completely aligned sequences at the other.

Preliminary experiments on these techniques show that they are effective in practice. We are working on incorporating them in TKB. The results from this study were presented in a comnference (please see paper attached).

Identifying Toxin Names and Interactions in Bio-Medical Abstracts

In addition developing the system we have embarked on text mining to identify new toxins from the available literature. A fully automated entity name extraction system, to identify toxins present in biomedical text has been developed. Our approach is based on identifying Sortal anaphors to extract proximal toxin names. We also extract protein-

protein interactions related to the toxins talked about in the given abstract. Our extraction system handles complex sentences and extracts multiple and nested interactions specified in a sentence. Deliverables are toxin name list extracted from PubMed abstracts for the query “Toxin Survey” and protein interactions related to these toxins.

Key Research Accomplishments:

1. We have built a sophisticated Toxin Knowledge Base.

2. This can be used to identify and store homolog information.

3. A powerful tool WinAgent developed at Stony Brook University can be used to retrieve and collect data from various sources and incorporated into TKB.

4. Various links have been incorporated into TKB for easy use.

5. The TKB now contains molecular, structural and other information for over 1000 toxins.

6. TKB now stores homolog information for more than 500 toxins with an easy to use software to view the structural model.

7. Text mining has been developed to identify new toxins from web site literatures.

Reportable outcomes

One paper presented in a conference.

1. Arvind Ramanthan, Mike Kifer, I.V. ramakrishnan, Arvind Ramanathan, Chang Zhao, S. Jayaraman and S. Swaminathan . Toxin Knowledge Base: A system for discovering bioengineered threats. Presented as a poster in ISMB conference in Detroit, June 2005.

2. Chang Zhao, Jalal Mahmud, I.V. Ramakrishnan and S. Swaminathan. Computing statistical profiles of active sites in proteins. SIAM Conference on Data Mining, 2006

3. Saikat Mukherjee, Chang Zhao and I.V. Ramakrishnan. Profiling Protein families from partially aligned sequences. Presented in a Computer science conference, 2006

Conclusion

TKB has thus provided an engineering solution to a widely acknowledged problem of analyzing information from various resources by combining several off-the-shelf software tools and developing an integrated work-flow that offers biologists with the ability to analyze the nature of toxins. It also provides information to users if a non-toxin protein can be potentially transformed into a toxin using simple substitution of amino-acid residues at their active sites. It is also the single largest resource on information regarding toxins, where in biologists can easily synthesize and disseminate knowledge about toxins.

Apart from our engineering processes, our current research effort focuses on the development of methods to classify toxins into families based on profiles (using profile based Hidden Markov Models [8]). These models take into account information about variations even across distant homologs and can thus identify remotely related proteins and toxins. We have successfully used these methods for comparing active sites in proteins.

It is also equally important to enrich the knowledgebase with more knowledge about toxins. But, this process is generally not very simple and often unintuitive, because, the identification of even a single new toxin can mean that a biologist has to potentially go through hundreds, perhaps even thousands of abstracts and scientific articles. Our text mining will solve this problem.

Future Plans

The plans presented here will be continued, if our application for funding to DTRA is successful.

1. Some toxins do not have any structure information available. In this case, we can instead get a model for the toxin from MODBASE. For example, Tityustoxin ts3 (accession number P01496) does not have a PDB entry in Swiss-Prot, but it has a highly reliable model in MODBASE.
2. There are toxins for which we cannot find the active site information in PDB, PDBSUM or LPC. We need to find alternative information sources of toxin active sites. Because active sites are often associated with structural pockets and cavities, one possible approach is to locate the active site with the help of CastP, which can provide identification and measurements of surface accessible pockets as well as interior inaccessible cavities.
3. Instead of building a model based on a pair-wise alignment and a toxin structure, we can build a model based on a multiple alignment of the homolog and several toxins to which it is homologous. This can help us to remove redundant models and improve the reliability of the models.

4. Hidden Markov Models have been used to build profiles of protein sequences of the same family. We want to extend the Profile Hidden Markov Model to the three-dimensional space and use it to model the active sites of toxins from the same family. Instead of comparing each active site with the target protein, we just need to compare each active site profile with the target protein.

5. The user will be allowed to submit a new toxin. The new toxins will be studied and it may become an entry in the TKB.

References

1. Using a Library of Structural Templates to Recognize Catalytic Sites and Explore their Evolution in Homologous Families. J. Mol. Biol. Vol 347 2005, pages 565-581
2. An algorithm for constraint-based structural template matching: application to 3D templates with statistical analysis. Bioinformatics Vol 19 no. 13 2003, pages 1644-1649
3. MODBASE, a database of annotated comparative protein structure models, and associated resources. Nucleic Acids Research Vol 32, Database issue D217-D222, 2004.
4. Nikeeta Julasana, Akshat Khandelwal, Anupama Lolage, Prabhdeep Singh, Priyanka Vasudevan, Hasan Davulcu, I. V. Ramakrishnan: WinAgent: a system for creating and executing personal information assistants using a web browser. Intelligent User Interfaces 2004: 356-357
5. Altschul, S.F., Gish, W., Miller, W., Myers, E.W. & Lipman, D.J. (1990) "Basic local alignment search tool." J. Mol. Biol. 215:403-410.
6. G.J. Kleywegt (1999). Recognition of spatial motifs in protein structures. J Mol Biol 285, 1887-1897.
7. Sali & T.L. Blundell. Comparative protein modelling by satisfaction of spatial restraints. J. Mol. Biol. 234, 779-815, 1993.
8. Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids, R. Durbin, S. Eddy, A. Krogh, G. Mitchison, Cambridge Publications, 1998.

Personnel in the Project

1. S. Swaminathan (PI)	Scientist	20% effort
2. S. Jayaraman	Sr. Research Associate	30% effort

Sub-contract to State University of New York at Stony Brook

1. Mike Kifer	Professor	10% effort
2. I.V. Ramakrishnan	Professor	10% effort

Two Ph.D. (50%) student and 12 M.S. students (all part time 20 to 50%)

Sub-contract to Arizona State University, Tempe, Arizona

1. H. Davulcu	Asst. Professor	10% effort

Seven M.S. students (part time)

Note: ASU was not part of no-cost extension period (1 August 2005 – 31 July 2006)

Appendix:

Three documents are appended.

1. Arvind Ramanthan, Mike Kifer, I.V. ramakrishnan, Arvind Ramanathan, Chang Zhao, S. Jayaraman and S. Swaminathan . Toxin Knowledge Base: A system for discovering bioengineered threats. Presented as a poster in ISMB conference in Detroit, June 2005.

2. Chang Zhao, Jalal Mahmud, I.V. Ramakrishnan and S. Swaminathan. Computing statistical profiles of active sites in proteins. SIAM Conference on Data Mining, 2006

3. Saikat Mukherjee, Chang Zhao and I.V. Ramakrishnan. Profiling Protein families from partially aligned sequences. Presented in a Computer science conference, 2006

Toxin Knowledge Base: A System

Michael Kifer, I. V. Ramakrishnan, Arvind Ramanathan, Chang Zhao

*Department of Computer Science, Computer Science Building,
Stony Brook University, Stony Brook, NY 11794.*

Abstract

Recent developments in recombinant DNA technology has given rise to the possibility of producing bioengineered pathogens like toxins and other products on scales that could make them into formidable weapons of bioterrorism. Yet another kind of threat is through chimeric molecules, where in the virulent domain of a toxin is hidden in a non-pathogenic protein. The Toxin Knowledge Base (TKB) has been established as a bioinformatics resource to tackle the problem of identifying potential bio-warfare agents as well as chimeric proteins. It is a tool that can be used to assimilate, synthesize, analyze and disseminate genomic and structural information on biological and potential biological warfare agents, identify and develop counter measures such as vaccines, antitoxins and inhibitors and also understand the mode of actions of these toxins at the cellular, sub-cellular, and molecular levels. The system has been designed using a novel workflow mechanism and is seamlessly integrated using an easy-to-use web based portal.

System Description

The system has been developed with an emphasis on the end user’s perspective in mind. Figure 1 shows the system architecture of TKB from that perspective. The Query interfaces comprise of a sophisticated inference engine, based on derivable knowledge in terms of the structures of toxins that exist within the database. This knowledge is used as a means to design the Trans-toxin workflow, whose schematic is shown in figure 2. Once a user inputs a sequence, a search against the sequences in the RCSB yields the answer to the question whether the input sequence has a structure. If the sequence does not have a structure, the input sequence is first ‘BLAST’ed against the TKB sequence information to determine whether it is possible to model the sequence using the structural information in the knowledge base. If so, it is passed on to the Modeller program, which provides a three dimensional model, which is then compared against the active site templates using SPASM and the results are tabulated to the end user.

The administrator’s interface is based on the fact that he spends most of his time in keeping the knowledge base up-to-date. The data acquisition and curation workflow is shown in figure 3.

The data for toxins lies embedded within several sources. Principal sources for the current work include public domain databases like SWISSPROT, EMBL and RCSB. Using a specially built tool called WinAgent, the data from these various sites are mined and assembled as XML data. This XML data is then processed using an XSLT to merge into the Toxin Knowledgebase. This provides a simple and extensible mechanism for mining toxin data from the internet. The tool created for this purpose (a modified version of WinAgent) is very easy to use, such that even a novice user can create powerful agents to mine for toxin data. This tool has been embedded within the design of the system such that it is transparent to the user and hence, the user can simply create agents by specifying a specific website and click on a few instances that he wants the agent to pick up (during the training phase), and then automatically execute the code to fetch similar sets of data from the site and then integrate it into the toxin repository.

This has allowed us to develop a simple, yet powerfully extensible system, that can be potentially applied to not only toxins, but also different classes of recognized proteins.

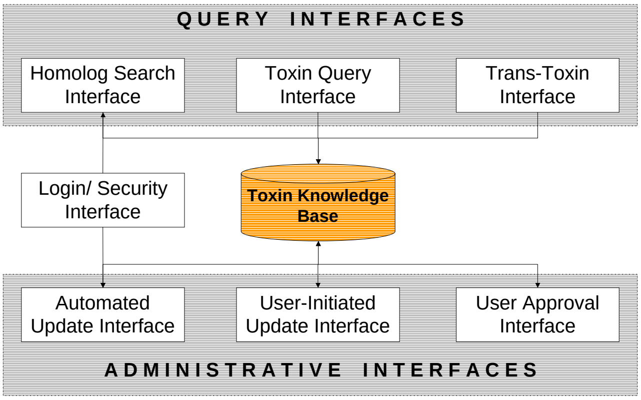


Figure 1: System Architecture

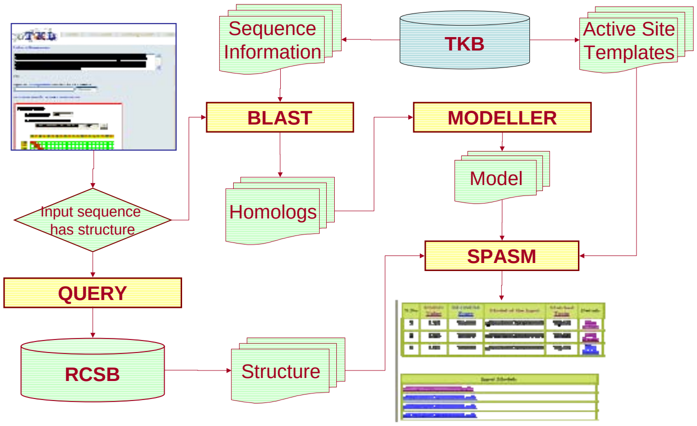


Figure 2: The Trans-Toxin Workflow

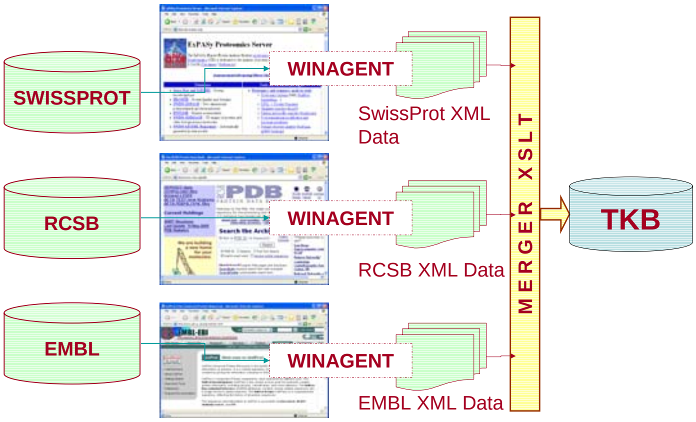


Figure 3: Data Acquisition Workflow

for Discovering Bio-Engineered Threats

Seetharaman Jayaraman, Subramanyam Swaminathan

Biology Department, Brookhaven National Laboratory,
Upton, NY 11973

Results

```
BoNT/E          PKINSFNYNDPVNDRTILYIKPGGCQEFYKSFNIMKNIWIIPERNVIGTTPQDFHPPTSL 60
THERMO          ---------------------------------ITGTSTVGVGRGVLGDQ---------- 17
                                                 *  .  :   *.*:*

BoNT/E          KNGDSSYYDPNYLQSDEEKDRFLKIVTKIFNRINNNLSGGILLEELSKANPYLGNDNTPD 120
THERMO          KNINTTYSTYYYLQDN--------------TRGDGIFTYDAKYRTTLPGSLWADADN--- 60
                ** :::*    ***.:              .* :. :: .   .    .. : . **

BoNT/E          NQFHIGDASAVEIKFSNGSQDILLPNVIIMGAEPDLFETNSSNISLRNNYMPSNHRFGSI 180
THERMO          QFFASYDAPAVDAHYYAGVTYDYYKNVHNR------LSYDGNNAAIRSSVHYS---QGYN 111
                : *   **.**: ::  *       **         :. :..* ::*..   *    *

BoNT/E          AIVTFSPEYSFRFNDNCMNEFIQDPALTLMHELIHSLHGLYGAKGITTKYTITQKQNPLI 240
THERMO          NAFWNGSEMVYGDGDGQTFIPLSGGIDVVAHELTHAVTDYTAGLIYQNE---SGAINEAI 168
                  .  ..*  :  .*.     :..   .: *** *:: .  ..    .:   :   *  *

BoNT/E          TNIRGTNIEEFLTFGGTDLNIITSAQSNDIYTNLLADYKKIASKLSKVQVSNPLLNPYKD 300
THERMO          SDIFGTLVEFYANK-------------NPDWEIGEDVYTPGISGDSLRSMSDPAKYGDPD 215
                ::* ** :* : .              *  :      *.   *  *  .:*:*      *

BoNT/E          VFEAKYGLDKDASGIYSVNINKFNDIFKKLYSFTEFDLRTKFQVKCRQTYIGQYKYFKLS 360
THERMO          HYSKRYTGTQDNG---GVHINSGIINKAAYLISQGGTHYGVSVVGIGRDKLGKIFYRALT 272
                 :. :*   :* .   .*:**.                     *   :  :*:  *  *:

BoNT/E          NLLNDSIYNISEGYNINNLKVNFRGQNANLNPRIITPITGRGLVKKIIRFCKNIVSVKGI 420
THERMO          QYLTPTSNFSQLRAAAVQSATDLYGSTSQEVASVKQAFDAVGVK---------------- 316
                : *. :    .      :  .:: *..::  . :  .: . *:

BoNT/E          R 421
THERMO          -
```

```
THERMO          --ITGTSTVGVGRGVLGDQKNINTTYSTYY------YLQDNTRGDGIFTYDAKYRTTLPG 52
NEP_HUMAN       GICKSSDCIKSAARLIQNMDATTEPCTDFFKYACGGWLKRNVIPETSSRYFAGESKHVVE 60
                ..:. :  .  :: : .  . . : ::      :   *: *.  :    * *   . :

THERMO          SLWADADNQFFASYDAPAVDAHYYAGVTYDYYKNVHNRLSYDGNNAAIR----SSVHYSQ 108
NEP_HUMAN       DLIAQIREVFIQTLDDLTWMDAETKKRAEEKALAIKERIGYPDKDEWISGAAVVNAFYSS 120
                .* *:  : *: : *  :         : :    :::*:.* .::  *      ...**.

THERMO          GYNNAFWNG--SEMVYGDGDGQTFIPLSGGIDVVAHELTHAVTDYTAGLIYQ-NESGAIN 165
NEP_HUMAN       GRNQIVFPAGILQPPFFSAQQSNSLNYGGIGMVIGHEITHGFDDNGRNFNKDGDLVDWWT 180
                * *: .: .   :  : ..: .. :  .*   *:.**:**.. *   .:  : :  .  .

THERMO          EAISDIFG--TLVEFYANKNPDWEIGEDVYTPGISGDSLRSMSDPAKYGDPDHYSKRYTG 223
NEP_HUMAN       QQSASNFKEQSQCMVYQYGNFSWDLAGGQHLNGIN-TLGENIADNGGLGQAYRAYQNYIK 239
                :  :. *   :   .*   * .*::. . :  **..   ..::* .  *:. :  :.*

THERMO          TQDNGGVHINSGIINKAAYLISQGGTHYGVSVVGIGRDKLGKIFYRALTQYLTPTSNFSQ 283
NEP_HUMAN       KNGEEKLLPGLDLNHKQLFFLNFAQVWCGTYRPEYAVNSIKTDVHSPGNFRIIGTLQNSA 299
                .:.:  :  . .: :*  :::. . .  *.     . :.: . .: . .  :  * : *

THERMO          LRAAAVQSATDLYGSTSQEVASVKQAFDAVGVK 316
NEP_HUMAN       EFSEAFHCRKNSYMNPEKKCRVW---------- 322
                : *.:. .: * ...::    ..: .: . .
```

Figure 4: Sequence comparison between botulinum neurotoxin serotype E Light Chain (BoNT/E), thermolysin (THERMO), and neprilysin (NEP_HUMAN), similar to a report presented in Toxin Knowledgebase. One can observe that, BoNT/E and THERMO are homologs; THERMO and NEP_HUMAN are homologs, and share the same active site motif (Zinc binding motif: HEXXH).

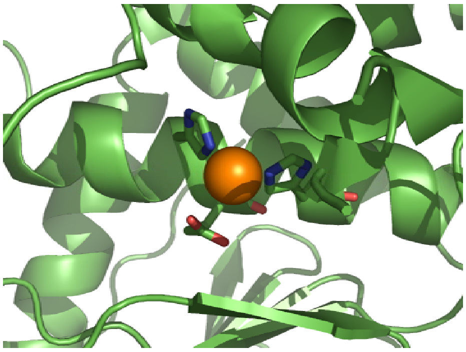


Figure 5 † (a): Active Site of botulinum neurotoxin E Light Chain (PDB: 1T3A), residues shown are H211, E212 and H215.

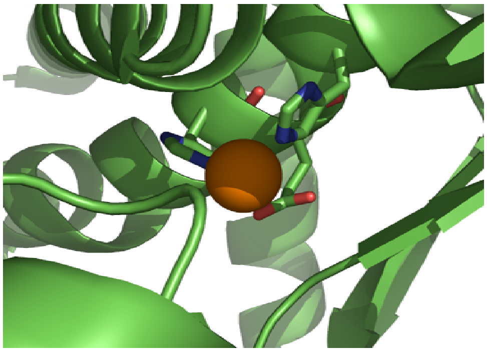


Figure 5 † (b): Active Site of human neprilysin (PDB: 1DMT), with residues H583, E584 and H587.

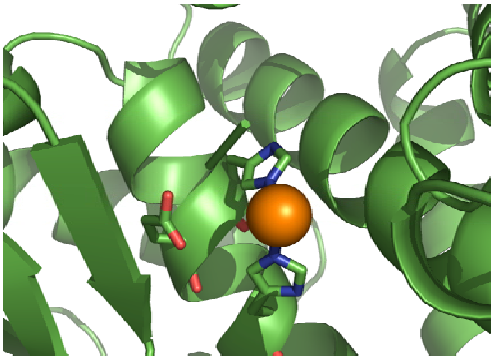


Figure 5 † (c): Active Site of thermolysin (PDB: 1TLP), residues H142, E143 and H146.

† Figures generated using PyMOL Molecular Graphics software, (http://www.pymol.org).

The initial results from using TKB are shown in figures 4 and 5. Figure 4 presents a sequence comparison of botulinum neurotoxin, thermolysin and neprilysin. As shown in the figure, all the three proteins share the same motif – HEXXH. However, one must note that botulinum neurotoxin serotype E and neprilysin are rather distantly related homologs. They do not show a high sequence similarity and hence one may conclude that the two may not share any significant structural or functional similarity.

If one follows the workflow illustrated in figure 2, the output from TKB will show that the active sites of the three proteins to be very similar. The arrangement of the residues at the active site and the coordination of the Zinc ion shows that all the three proteins not only share the same motif, but also that these proteins could be functionally similar.

Conclusions and Future Work

The current work represents a successful prototype for relating sequence, structure and functional aspects of proteins. Sequence based comparison methods although valuable, can allow detection of related families of proteins, but in order to relate the functions of proteins better, better structure based methods are necessary. In the future, the authors plan to develop three dimensional structural profiles, that would be used to identify remote homologs, that may be functionally related to various proteins. It is also planned to expand the knowledgebase and allowing the bioinformatics community to use this web-based service.

References

1. Nikeeta Julasana, Akshat Khandelwal, Anupama Lolage, Prabhdeep Singh, Priyanka Vasudevan, Hasan Davulcu, I. V. Ramakrishnan. WinAgent: a system for creating and executing personal information assistants using a web browser. Intelligent User Interfaces 2004: 356-357.

2. Altschul, S.F., Gish, W., Miller, W., Myers, E.W. & Lipman, D.J. (1990). Basic local alignment search tool. J. Mol. Biol. 215:403-410.

3. G.J. Kleywegt (1999). Recognition of spatial motifs in protein structures. J Mol Biol 285, 1887-1897.

4. A. Sali & T.L. Blundell. Comparative protein modelling by satisfaction of spatial restraints. J. Mol. Biol. 234, 779-815, 1993.

Acknowledgements

The authors would like to thank Sridhama Vempaty, Padmavathy Malligarjunan, Jigar Mehta, Darshan Ramavat, Venkat Reddy Nukala, Sheetal Shah, Mayur Shetty, Debashish Rawal and Pooja Virkud for their help in developing the system. The grant from US Army Medical Research Acquisition Activity (Contract No.: **DAMD17-03-1-0520**) is also gratefully acknowledged.

Computing Statistical Profiles of Active Sites in Proteins

Chang Zhao, Jalal Mahmud, and I.V. Ramakrishnan
Computer Science Department
Stony Brook University
Stony Brook, NY 11794-4400
{changz, jmahmud, ram}@cs.sunysb.edu

Subramanyam Swaminathan
Biology Department
Brookhaven National Laboratory
Upton, NY11973-5000
swami@bnl.gov

Abstract

Active sites in proteins are three dimensional substructures that cause them to perform their function. The problem of finding substructures in a protein that are "similar" to the active sites of another protein has several important applications in biological sciences such as drug design, genetic engineering, and diagnostic tools for analysis of genetically engineered pathogens. Active sites can be grouped into families whose members are related by similarity of their functions. Since similar sites exhibit variability in their physico-chemical and structural features, statistical profiling methods capture the shared features robustly in the presence of such variations. In this paper, we adapt Profile Hidden Markov Models (PHMMs) that have been successfully used for analyzing biological sequences, to statistically profile active site families. Since PHMMs can only profile one dimensional sequences, we develop a serialization of the three dimensional active sites that captures certain shared physico-chemical and geometric features of the family. PHMM parameters are learnt using these serialized sequences. While traditional PHMM learning algorithms deal with discrete physico-chemical feature only, we expand it to include geometric features drawn from a continuous probability distribution. Experimental results with our PHMM based method for profiling active sites suggest that it is effective in practice.

1 Introduction

Proteins are essential to the structure and function of all living cells and viruses. Understanding the function of a protein is fundamental for gaining insight into many biological processes. Technically, proteins are amino acid chains that fold into unique three-dimensional structures that cause them to function. In particular, within the protein structure are key areas called *active sites* and biochemical reactions at these sites with other proteins or other chemical substances cause the protein to perform a function of one type or another.

A problem of significant importance in computational biology is this: *Are active sites of different proteins similar?* i.e., do they share similar physico-chemical and geometric properties. Active sites with such shared properties may perform similar functions. Answer to the aforementioned similarity question drives a number of important biological applications. For instance it can be used to predict the function of a protein with a substructure similar to the active site of another protein whose function is known. Another important application is toxicology tools such as the Toxin Knowledge Base (TKB) system that we have developed [9, 15, 21], for automated diagnosis of bioengineered pathogens. In such pathogens the virulent domains of toxins can be hidden in otherwise non-toxic proteins. Specifically, the active site of a non-toxic protein that is similar to that of a toxin, has the potential to become toxic by suitably altering the *residues*[1] in the site.

State-of-the-art techniques for determining active site similarity are exemplified by the SPASM tool [10, 22]. Its inputs include a protein's structure; the 3-D coordinates of the residues in the active site of another protein whose function is known, substitutions for these residues and a RMSD (root mean square distance) cutoff value. SPASM attempts to identify 3-D substructure(s) of the former protein that are isomorphic to the active site within the specified RMSD cutoff.

There are two problems with the pairwise similarity testing approach embodied in SPASM. Firstly, although there are general guidelines for choosing RMSD values such as "If you use only a few residues (3-5), an RMSD less than one Å tends to be obtained for similar arrangements of residues,"[2] in general it is a laborious trial and error process. However, the more serious problem is that similarity

[1] Informally, the residues are the elements joined together in the amino acid sequence.

[2] Å denotes an angstrom which is the distance measure between atoms. One angstrom is 1.0×10^{-10} meters.

tests are done separately with one active site at a time. Consequently, it does not exploit the common physico-chemical and structural features that can exist amongst the *family* of active sites of proteins. A family here means that the active sites of all of its members exhibit similar functionality and can also include evolutionarily unrelated proteins that share no overall sequence or fold similarities. Pairwise comparisons may use features that may not be common to all the family members and hence can fail to identify family members, especially "remote"[3] members. For instance, SPASM fails to find the similarity between the active sites of BOVINE RIBONUCLEASE (PDB ID: 3RN3)[4] and a variant of RIBONUCLEASE (PDB ID: 1RBC) for reasonable RMSD cutoffs because atoms not directly related to the protein's function differ a lot in these two structures. Note however that a "*profile*" of the common features in a collection of active sites belonging to a family would have revealed the irrelevance of such atoms and hence would have been excluded as a shared feature. So a principal benefit of profile based methods is that they capture the essential features shared by all of the family members thereby making it possible to determine the similarity of remote members.

Automated construction of active site family profiles to discern common features is a fairly unexplored problem. In this paper we formulate a solution to this problem inspired by the successful profile-based search methods for homologous protein sequences[5] [16].

Note that physico-chemical and structural features of similar active sites may exhibit some degree of variability. So similarity notions rooted in statistics can serve as a robust framework for profiling the active sites of a family.

Profile Hidden Markov Model (PHMM), a statistical learning technique used in profile based sequence homology search methods, has been shown to be very effective for capturing sequence similarity [5]. Several software tools based on PHMM have also been developed [6, 7, 8]. PHMMs are in essence HMMs adapted for profiling sequence similarity. HMMs are a class of probabilistic automata used in a number of applications, especially sequence labeling problems such as recognizing words in digitized speech [17], attribute data extraction from text sequences [3] and biological sequence analysis.

We adapt PHMM for profiling the three dimensional active sites in proteins. To begin with, the adaptation requires choosing a representative set of active site features. Whereas only residue types (such as Histidine, Glutanmate, etc) are used as features in PHMMs for prtein sequences we will now have to contend with the structural (i.e., geometric) features of active sites also. So in addition to using the atoms' types in the active site residues we also use their distances from their center of mass as the structural features. Furthermore these distances are assumed to be drawn from a probability distribution. Next we adapt the training phase of PHMM to learn the parameters of this distribution and finally modify the scoring phase to assign a similarity score to the input data.

We have implemented a prototype of our adaptation of PHMM. Using this system we can determine the similarity of protein substructures with the family profile of active sites. In fact our system determined that the active sites of BOVINE RIBONUCLEASE (PDB ID: 3RN3) and a variant of RIBONUCLEASE (PDB ID: 1RBC) are similar which SPASM had failed to do as was mentioned earlier. Templates for capturing active site features are described in [20] and more recently in [19]. However these templates are constructed manually. In contrast our system learns these features automatically. Finally and most importantly in our approach there is no need to manually figure out RMSD cut-offs as is needed to be done when using SPASM.

The rest of the paper is organized as follows. Section 2 presents an overview of protein active sites and PHMMs to set the context for understanding the rest of the paper. Section 3 provides details of our adaptation of PHMM for active site profiling. Section 4 presents experimental results of our approach. Related work appears in Section 5 and conclusions in Section 6.

2 Technical Preliminaries

In this section we review the technical background needed to understand our technical approach. In particular the review focuses on active sites in proteins and PHMM.

2.1 Protein Active Site

The building blocks of proteins are twenty amino acids. Examples of these include Alanine, Valine, Histidine, Glycine, etc. They are usually referred to by their symbolic (3-letter and 1-letter) abbreviations e.g., the 3-letter ALA or the 1 letter A for Alanine, VAL or V for Valine and so on.

All of the twenty amino acids have in common a central carbon atom (C_α) to which are attached a hydrogen atom, an amino group (NH_2), and a carboxyl group ($COOH$). The rest of an amino acid, which is called the *side chain*, is different for different amino acids. These terms are illustrated in Figure 1. Amino acids are joined end-to-end to form a polypeptide chain when the carboxyl group of one amino acid condenses with the amino group of the next to eliminate water, as shown in Figure 2. The repeating units in the polypeptide chain are called *residues*. In Figure 2 the

[3] These are active sites that have few features in common with the other family members.

[4] PDB –`http://www.rcsb.org` – is the Protein Data Bank of 3-D protein structures uniquely indexed by an ID

[5] A protein sequence is simply a linear string of amino acids that constitute the primary structure of a protein. Sequences that are similar are referred to as homologues.

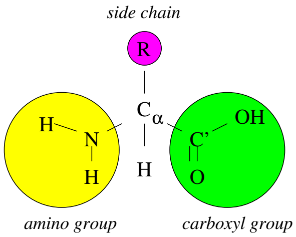


Figure 1. Schematic Diagram of an Amino Acid

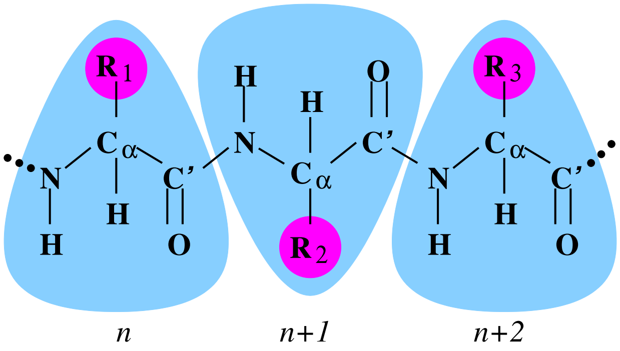


Figure 2. Polypeptide Chain

elements within each "shaded triangular" area correspond to a residue. A residue is usually referred to by its name or abbreviation followed by its position in the chain. For example, H233 refers to the 233rd residue in a chain, which is a histidine.

The polypeptide chain of a protein folds in space to form the three-dimensional structure of the protein. The folding of the polypeptide chain typically creates a crevice or cavity on the protein surface. This crevice, called an *active site*, contains a set of residue side chains which might be far apart in the polypeptide chain. They are brought together in the 3-D structure and are disposed in such a way that they can make noncovalent bonds only with certain partners, which can be a protein, DNA, metal ion, etc. The 3-D structure of a protein, especially the localized structure of its active site, determines the functional properties of the protein. Figure 3 sketches the formation of an active site. Note that a protein can have several active sites. By examining the interaction of a protein and its binding partner, the protein's active site can be identified. Alternatively, active sites can be inferred by computational tools such as MOE Active Site Finder [23] and Q-SiteFinder [13].

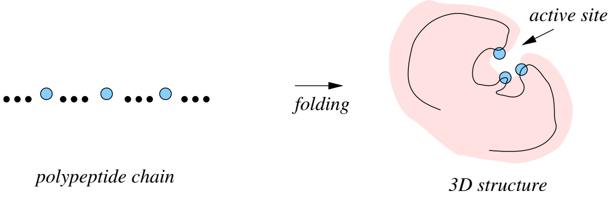


Figure 3. Formation of an Active Site

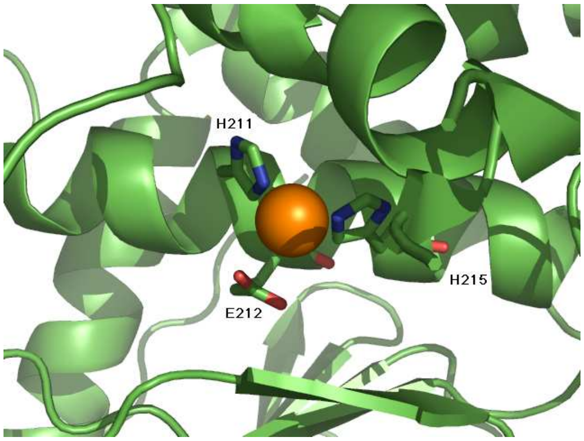


Figure 4. Active Site of Botulinum Neurotoxin Serotype E(PDB ID: 1T3A)

Figure 4 shows the active site of butolinum neurotoxin serotype E. It contains three residues: H211, E212, and H215, represented by the sticks in the figure. It determines that this protein has the function of binding a zinc ion which is represented by the ball in the figure.

2.2 Profile Hidden Markov Model

A PHMM is a statistical learning-based technique for modeling DNA and protein sequences families. The underlying principles of PHMMs are based upon the mathematics of Hidden Markov Models [17] which have found wide applicability in sequence analysis tasks. An HMM is a probabilistic finite state automaton defined by a set of states, a set of state transitions with probabilities assigned to them and a set of observation symbols that are emitted in a state with certain probabilities. The sequence of states corresponding to a visible observation sequence is "hidden" and hence has to be estimated.

PHMMs extend the traditional notion of HMMs to model biological sequence families. In this section, we briefly review PHMMs and their application in modeling biological sequence families. A more detailed discussion can be found in [5]. We remark that PHMMs for modeling DNA and protein sequences mainly differ in the domain of emission symbols used. So without loss of generality our review will describe PHMMs for protein sequences only.

Protein sequences typically come in families. Members of a family have a common ancestor and normally main-

```
HBA_HUMAN        ...VGA--HAGEY...
HBB_HUMAN        ...V----NVDEV...
MYG_PHYCA        ...VEA--DVAGH...
GLB3_CHITP       ...VKG------D...
GLB5_PETMA       ...VYS--TYETS...
LGB2_LUPLU       ...FNA--NIPKH...
GLB1_GLYDI       ...IAGADNGAGV...
                    ***  *****
```

Figure 5. A Segment from the Multiple Alignment of 7 Globin Protein Sequences

tain the same or related function. Although they have diverged during evolution through insertions and deletions, their functional residues are usually conserved. A multiple alignment of family members reveals the relationship among them. For example, in Figure 5 which is a segment of the multiple alignment of seven globin protein sequences taken from [5], it is obvious that residues in some columns are more conserved than in others. A simple rule to decide whether a column is conserved is that if more than half of the sequences have a residue instead of a dash present in the column, then that column is conserved. In Figure 5, the columns marked with stars are conserved. The two non-starred residues in GLB1_GLYDI correspond to insertions. If a sequence has a dash in a conserved column, then it has undergone a deletion.

PHMMs are HMMs whose structures are specialised to capture such conserved residues as well as insertions and deletions in sequence families. Figure 6 shows the structure of a PHMM. The structure has a *Begin* state and an *End* state, denoted B and E respectively in the figure, and a sequence of columns of states between B and E. Each column, from 1 to n, has three states - a *match*, *insert*, and *delete* state. These are denoted by M_i, I_i, and D_i respectively for the i^{th} column. Intuitively, match states correspond to conserved residues among sequences while insert and delete states correspond to divergence in sequences from a common ancestor due to insertions and deletions respectively. The insert state I_0 corresponds to insertions before the first matching residue in sequences. Observe from Figure 6 that the structure of the model is parameterized only by the length of the model, i.e., the number of columns of states.

The transitions in the model structure are fixed and corresponds to the underlying semantics of matches, insertions, and deletions. In particular, a match state M_i can make a transition to M_{i+1}, I_i and D_{i+1} respectively. An insert state I_i has transitions to M_{i+1}, D_{i+1}, and to I_i itself. A delete state D_i has transitions to M_{i+1}, D_{i+1}, and I_i. These state transitions are also shown in Figure 6.

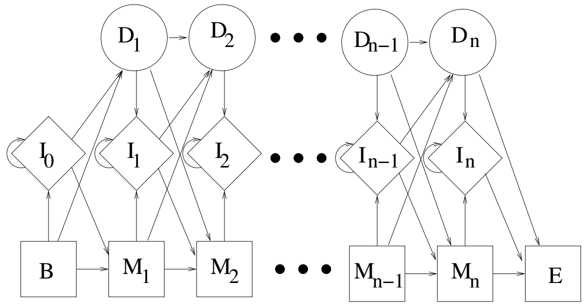


Figure 6. PHMM structure

For protein sequences, the emission symbols are the twenty amino acids. Match and insert states emit residues while delete states are non-emitting silent states. The non-emission of residues from delete states conforms to the semantics of these states – a residue in the representation of the family is not observed in an individual sequence. The begin and end states define the start and end markers of the model and consequently they do not emit residues.

The parameters of a PHMM are usually learned from a set of sequences known as members of a family. When the alignment of the sequences is given, computing the model probabilities reduces to smoothed maximum-likelihood parameter estimation using the frequency counts of transition and emission events. Figure 7 (taken from [5]) is a PHMM of length 8. Emission probabilities are shown as bars opposite the different amino acids for each match state, and the values of transition probabilities are indicated by the thickness of the lines. Th self looping transitions on the insert states are probability values given as percentages. The emission probabilities are uniformly distributed among 20 amino acids for all the insert states except I_3 where the emission probabilities are 0.09 for amino acid A and D and 0.045 for all the others.

When the alignment of the training sequences is unknown, learning PHMM parameters is done with Baum-Welch's [2] iterative algorithm which is a special case of the more general Expectation-Maximization (EM) algorithm [4]. Starting from initial parameter values, the algorithm terminates after a fixed number of iterations or after a local maximum has been reached. In each iteration, the current parameter values are used to first compute transition and emission expectations (E step) which are then subsequently used to generate the best possible parameter values for the next iteration (M step).

Given a PHMM model M profiling a family S and an input protein sequence x the model determines if x is a member of S, i.e., is it similar to the members of S profiled by M? At a high level this is done as follows: First, the best path (i.e., state sequence) is computed using the well known Viterbi algorithm [17]. In particular, the

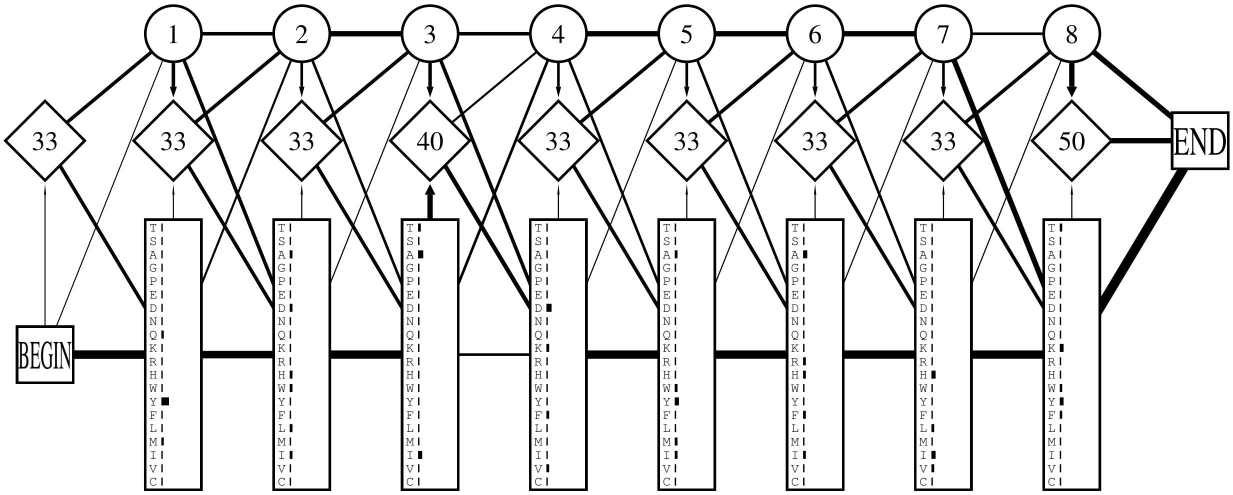


Figure 7. An Example PHMM

Viterbi algorithm efficiently computes a state sequence y' that maximize the conditional joint probability $P(x, y|M)$, i.e., $y' = arg \max_y P(x, y|M)$. For example, the best path for the sequence "VGAHAGEY" and the model in Figure 7 is found to be $Start \rightarrow M_1 \rightarrow M_2 \rightarrow M_3 \rightarrow M_4 \rightarrow M_5 \rightarrow M_6 \rightarrow M_7 \rightarrow M_8 \rightarrow End$. Next, we compute $p(x, y'|M')$ where M' is a random model that is identical to M in length and transition probabilities. However, the emission probability for each emission symbol a is independent of the states, i.e., for all match and insert states a always occurs with the same frequency q_a. A choice for q_a is the frequency of the amino acid a occurring in a standard sequence database such as SWISS-PROT [24]. Finally we compute the base 2 log-odds ratio $log(\frac{P(x,y|M)}{P(x,y|M')})$ called the *bit score*. If this score falls above a threshold then x is said to be a member of S. The threshold is a global value and details on how it is determined appears in [25].

3 Profiling Active Sites with PHMM

Note that active sites with similar functions can exhibit variability in their physico-chemical and geometrical configurations. So any technique for profiling active sites should factor in such variations. PHMMs have been used for biological sequence analysis with a high degree of success since they can statistically capture commonalities and variations among sequences that have evolved from a common ancestor. Thus they have the potential to serve as a robust framework for profiling active sites also.

However, adapting PHMMs for this problem is not entirely straightforward. Let us examine the underlying issues. Firstly, observe that PHMM is a sequential model in the sense that it was developed to handle protein sequences which are simply 1-D strings of amino acids. On the other hand active sites are 3-D structures. So the immediate problem is one of serializing these 3-D structures in such a way that salient aspects of their physico-chemical and geometric properties are still retained. Secondly, each state in a traditional PHMM emits only one discrete symbol (i.e., an amino acid) at a time. For active sites these emissions must include both physico-chemical features such as the discrete valued residue types as well as geometric features. So emissions are tuples ranging over the physico-chemical and geometric feature set. A robust description of geometric configurations of active sites is best done using continuous measures. Hence in contrast to traditional PHMMs where only discrete probabilities of emission symbols are estimated we will now need to estimate the joint distribution of physico-chemical and geometric features. In the rest of this section we describe how we address these issues.

3.1 Serializing Active Sites

Since PHMM is a sequential model the task now is to identify a set of 3-D features and serialize them. This serialization will represent the observation sequence corresponding to an active site.

The primary issue in serialization is inventing an ordering for the sequence. For primary protein sequences of amino acid chains this is simply the position of the residue in the chain. For 3-D active sites there is no such obvious ordering. Let us first examine the desiderata for such an ordering. Ideally, if a and b are two conserved atoms in one active site, a' and b' are atoms in another active site corresponding to a and b, respectively, then the order of a and b in the serialized sequence derived from the former active site should be consistent with that of a' and b' in the sequence derived from the latter. A candidate for

Atom Name	X-Cord	Y-Cord	Z-Cord	Distance
N	36.729	107.613	20.276	2.427
CA	35.813	107.722	21.395	1.133
C	36.031	109.051	22.149	1.732
O	37.157	109.446	22.496	2.519
CB	35.875	106.405	22.220	1.016
CG	34.949	106.290	23.394	1.622
OD1	33.858	106.833	23.442	2.169
OD2	35.341	105.659	24.387	2.602

Table 1. Example Active Site Atoms

such an ordering is the distance of the atoms in the active site from their center of mass. Given a set of n atoms with coordinates $(x_1, y_1, z_1), (x_2, y_2, z_2), \ldots, (x_n, y_n, z_n)$, their center of mass is the expression:

$$(\frac{1}{n}\sum_{i=1}^{n} x_i, \frac{1}{n}\sum_{i=1}^{n} y_i, \frac{1}{n}\sum_{i=1}^{n} z_i)$$

In other words the center of mass is the average over each of the coordinate positions of the atoms. For illustration, suppose an active site contains only one residue D260 with atoms whose coordinates are listed in the first four columns of Table 1. The 3-D coordinate of their center of mass is $(35.719, 107.377, 22.470)$. Distances of each atom from the center of mass are shown in the last column of Table 1. The ordering of atoms arranged in ascending order of their distances from the center of mass is: < CB, CA, CG, C, OD1, N, O, OD2 >.

To capture physico-chemical feature, we adopt the atom classification in [14] which classifies all non-hydrogen atoms in proteins into 40 classes according to the atom location (side-chain or backbone), connectivity, and chemical nature. We denote the atom type by $ResidueName.AtomName$, which can be unambiguously mapped to an atom type in [14]. For example, the type of the first atom in Table 1 is represented by D.N.

As far as geometric feature is concerned, an obvious idea is to use an atom's 3-D coordinate. However, the coordinates of atoms from two active sites are comparable only after those active sites are superposed. Typically, superposing algorithms take two point sets with each point represented by its (x, y, z) coordinate, and perform rigid transformations such as translation and rotation to minimize the RMSD of these two point sets. Since these points are assumed to be typeless, any two points are always superposable. But the problem here is that superposed positions may not be compatible with the atom types at those positions (e.g., in general nitrogen and oxygen atoms cannot be superposed). There are tools such as SPASM [22] that allow users to define superposable atom types. The main problem with this is that knowledge about what are superposable atom types varies from family to family. A desiderata of geometric feature is that it be preserved under serialization. Features that use relative instead of absolute positions can satisfy such a requirement. Observe that distances of atoms to their center of mass are relative quantities and hence can serve as a geometric feature.

```
2DHC    <D.OD2,2.91>     -           -
1CHO    <D.OD2,3.08>     -     <H.CB ,3.17>
1ACE    <E.OE1,2.32><H.CD2,2.32><H.CB ,2.45>
             *                       *

2DHC         -     <H.CA ,3.10><H.O  ,3.20>
1CHO    <D.CG ,3.22><H.CA ,3.63>     -
1ACE         -     <H.CA ,3.31><E.CD ,3.46>
                         *           *

2DHC    <H.C  ,3.46><D.CG ,3.47><D.OD1,3.50>
1CHO    <H.C  ,4.06>     -           -
1ACE    <H.C  ,3.94>     -     <E.OE2,4.06>
             *                       *
```

Figure 8. A Segment of a Multiple Alignment

In summary, our feature set is the pair $\langle AtomType, Distance\_To\_CenterOfMass \rangle$, where the first element is the physico-chemical feature and the second is the geometric feature. The general form of an observation sequence corresponding to an active site following serialization using our feature set will be: $\langle t_1, d_1 \rangle$, $\langle t_2, d_2 \rangle$, ..., $\langle t_n, d_n \rangle$ where n is the number of atoms in the active site, t_i is the atom type and d_i is the distance to the center of mass for i = 1, ..., n, and $d_i < d_i + 1$ for i=1, ..., n-1. For our example active site, it is ⟨D.CB,1.016⟩, ⟨D.CA,1.133⟩, ⟨D.CG,1.622⟩, ⟨D.C,1.732⟩, ⟨D.OD1,2.169⟩, ⟨D.N,2.427⟩, ⟨D.O,2.519⟩, ⟨D.OD2,2.602⟩.

3.2 PHMM for Active Sites

When observation sequences of multiple active sites with similar function are put together, one can identify which atoms are conserved by aligning them. Figure 8 shows a segment of the alignment of three similar active sites[6], namely, acetylcholinesterase (PDB ID: 1ACE) with residues S200, E327, and H440 ; chymotrypsin (PDB ID: 1CHO) with residues H57, D102, and S195; haloalkane dehalogenase (PDB ID: 2DHC) with residues D124, D260, and H289. Although their constituting residues are different, all of them perform similar catalytic function.

This alignment reveals that the consensus sequence has six atoms (see columns marked by '*' in the figure). The atoms appearing in the non-starred columns are insertions. Observe also that the sequence of 2DHC goes through a deletion between the first match and the third match; 1CHO

[6] Because of width constraints the sequences in the figure run over to multiple lines.

goes through two deletions: one between the third match and the fifth match and the other after the fifth match.

We can learn the PHMM parameters (transition and emission probabilities) from such multiple alignments. However, it is labor intensive to come up with such a multiple alignment. One can also learn these parameters from unaligned sequences. First, the number of match states (i.e. the length of the PHMM) is estimated by taking the average length of the training sequences. Then the Baum-Welch algorithm is applied to estimate the transition probabilities and emission probabilities.

We adapt this process for learning PHMM parameters from training data consisting of unaligned serialized active site sequences belonging to a family. First, we estimate the length of the PHMM from the training sequences. This is the average length of the sequences. For example, the average length for the sequences in Figure 8 without the dashes is six.

To learn the other two PHMM parameters, we modify the Baum-Welch algorithm. Since emission symbols are pairs $\langle atomtype, distance\rangle$, we will need to compute the joint distribution of these pairs for each state. Making the standard independence assumption done in HMMs, namely, that the random variables in the joint distribution are independent, the probabilities of the atom types and their distances are computed separately. Let us define the probability of atom type t in a state as $P(t)$ and the probability of the distance d from center of mass as $P(d)$. We calculate the emission probability $P(b)$ of the emission symbol $b = \langle t, d\rangle$ to be $P(t) \times P(d)$.

The distance from the center of mass is a continuous feature. We assume that its probability distribution is generated by a multivariate Gaussian distribution whose probability density function is:

$$\frac{e^{-(\frac{(d-\mu)^2}{2\sigma^2})}}{\sigma\sqrt{2\pi}}$$

where d is the distance, μ is the mean and σ is the standard deviation of distances to the center of mass. Suppose the distances to the center of mass from atoms that are emitted by a state are $d_1, \ldots, d_m$. We compute μ and σ at this state using the expressions:

$$\mu = \frac{1}{n}\sum_{i=1}^{n} d_i$$

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(d_i - \mu)^2 + \epsilon}$$

The small constant ϵ is added so that σ is always positive even when $n = 1$.

Recall that we need 42 parameters to describe the emission distribution for each state. Forty of these parameters correspond to the emission probabilities of the 40 atom types and they must sum up to 1. The remaining two are μ and σ that represent the distribution of the distances of atoms emitted from the state to their center of mass.

For a set of unaligned sequences, Baum-Welch algorithm iteratively updates the parameters of the model to increase the overall probability of the set of training sequences to be generated by the model. We modify the Baum-Welch algorithm to take into account the new emission parameter set and the joint emission probability. At each step of iteration, we calculate the individual probabilities of atom type and distance from center of mass and multiply these probabilities to get the joint probability. For a family of observation sequences of active sites, this modified Baum-Welch algorithm is used to estimate the parameters of the PHMM that profiles this family.

Armed with a PHMM M trained on a family S of serialized active site sequences we can now answer questions about similarity of active sites. To determine if a protein has substructures similar to the active sites in S we proceed as follows: First we find candidate substructures in the protein structure. This can be done with tools such as MOE Active Site Finder [23] and Q-SiteFinder [13]. Then a serialized observation sequence is generated for each candidate substructure. Those are the candidate observation sequences for the protein. For each such observation sequence, we apply the Viterbi algorithm to compute the the probability of its most likely path in the PHMM M. To compute the probability of observing a pair $\langle t, d\rangle$ at a state, the Viterbi algorithm computes the probabilities of observing atom type t and distance d separately using the emission distribution parameters of that state, and then multiply them to get the emission probability of the pair.

The step that remains is computation of the log-odds ratio (see Section 2.2). For the PHMM M we define a random model M' whose length and transition probabilities are identical to those in M. The emission parameters are assumed to be uniform for all the insert and match states. These state-independent parameters are computed as follows:

1. The emission probability for atom type a is $\sum_{r\ where\ a\in r} \frac{q(r)}{num\ of\ atoms\ in\ r}$, where r is a residue and $q(r)$ is the frequency of r (see Section 2.2).

2. Randomly sample substructures from PDB, each of which contains the same number of residues as the training examples.

3. For each such substructure, compute the center of mass and the distances of the atoms to this center.

4. Compute the mean μ and the standard deviation σ over all distances and over all substructures.

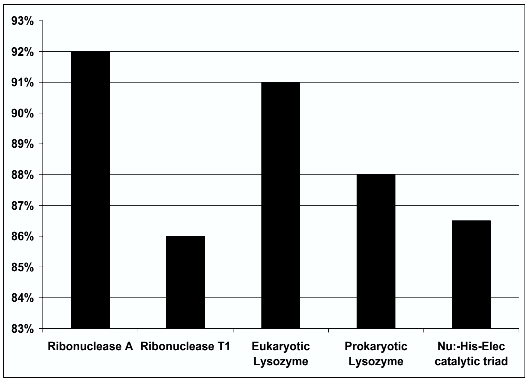


Figure 9. Precision Performance of Protein Families

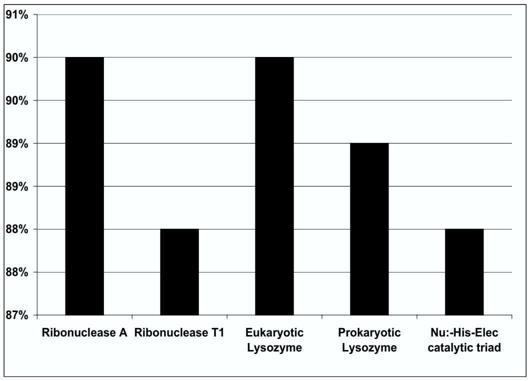


Figure 10. Recall Performance of Protein Families

Armed with M and M', we can compute the bit score of an observation sequence in M and decide similarity as was described in Section 2.2.

4 Evaluation

We implemented our PHMM-based profiling of active sites. In this section we report on its experimental performance. It is organized into the following subsections: The experimental setup for the evaluation; the performance metrics measured; the experimental results; and discussion of the results.

4.1 The Experimental Setup

The evaluation was conducted over different sets of protein families detailed below.

Protein Families

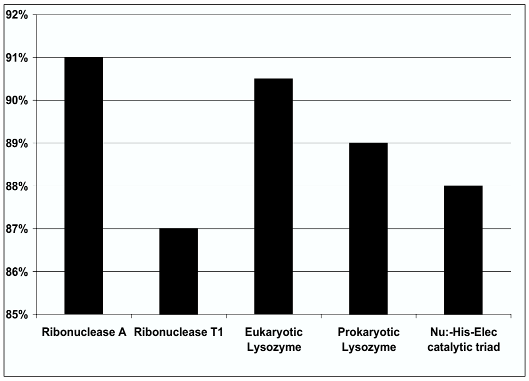


Figure 11. F-Measure Performance of Protein Families

We developed PHMM profiles for five different protein families, namely, *Ribonuclease A, Ribonuclease T1, Eukaryotic Lysozyme, Prokaryotic Lysozyme, Nu:-His-Elec catalytic triad.* The Nu:-His-Elec family is further divided into five subfamilies according to the residues that comprise the catalytic triads, which are Ser-His-Asp, Ser-His-Glu, Asp-His-Asp, Ser-His-Trp, and Cys-His-Asn. They are denoted by sub1, sub2, sub3, sub4, and sub5, respectively. We also built profiles for these five subfamilies.

Training and Test Data

We used 35, 34, 30, 35 and 153 members respectively of Ribonuclease A, Ribonuclease T1,Eukaryotic Lysozyme, Prokaryotic Lysozyme and Nu:-His-Elec catalytic triad families. For profiling the subfamilies we used 30, 35, 25, 31, 32 members of sub1, sub2, sub3, sub4, and sub5 respectively.

The active sites per family were divided into two mutually exclusive training and test sets. The active sites of a family included in the test set associated with the family were labeled as positive test examples. For each family, we augmented its test set with a subset of active sites belonging to other four families. These augmented active sites were labeled as negative test examples.

Statistics associated with the experimental data used are listed in table 2 for the five families and in table 3 for the subfamilies of Nu:-His-Elec catalytic triad.

From these statistics observe that on the average we used around 43 and 44 active sites respectively for training and testing each family.

We built a separate PHMM per family. The parameters were learnt using the training set associated with the family. The global threshold for the log-odds ratio was set to 0.

Table 2. Data Statistics for Different Protein Families

Protein Families	Average No of Active Site Atoms	Size of Training Set	Size of Test Set	No of Positive Examples in the Test Set	No of Negative Examples in the Test Set
Ribonuclease A	20	25	30	10	20
Ribonuclease T1	24	24	35	10	25
Eukaryotic Lysozyme	29	30	20	8	12
Prokaryotic Lysozyme	21	35	31	15	16
Nu:-His-Elec catalytic triad	24	103	105	40	65

Table 3. Data Statistics for Subfamilies of Nu:-His-Elec Catalytic Triad

Protein Families	Average No of Active Site Atoms	Size of Training Set	Size of Test Set	No of Positive Examples in the Test Set	No of Negative Examples in the Test Set
sub1	20	18	32	12	20
sub2	22	20	35	15	20
sub3	26	15	22	10	12
sub4	28	18	25	13	12
sub5	24	21	21	11	10

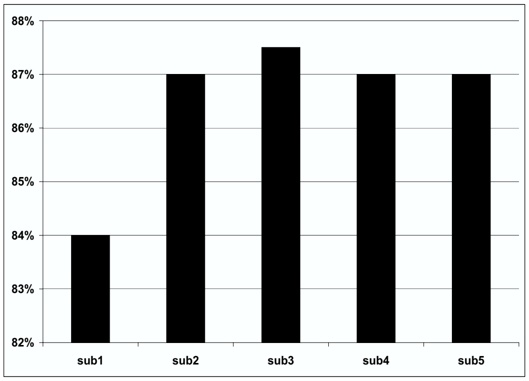


Figure 12. Precision Performance of Protein Sub Families of Nu:-His-Elec Catalytic Triad

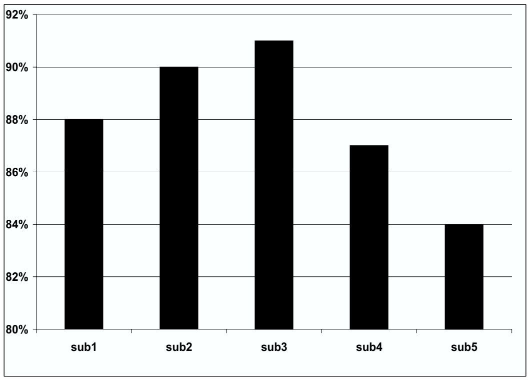


Figure 13. Recall Performance of Protein Sub Families of Nu:-His-Elec Catalytic Triad

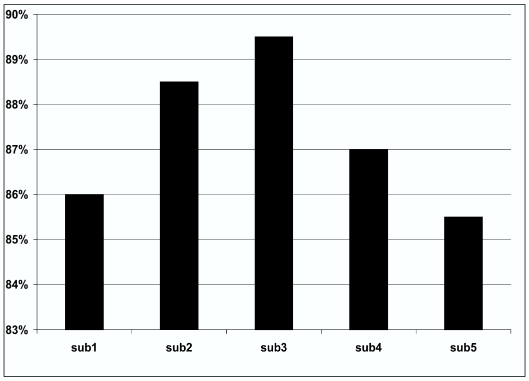


Figure 14. F-Measure Performance of Protein Sub Families of Nu:-His-Elec Catalytic Triad

4.2 Performance Metrics

We evaluated the PHMM with respect to three performance metrics: recall, precision and f-measure [7] using the test data set constructed above.

Observe that an active site in the test data for a family was uniquely labeled as a positive or negative test example. These labels are used to classify the similarity results produced by PHMM on the test data into true positives, false positives, true negatives and false negatives. Based on these classifications the recall/precision/F-measures are directly computed from their definitions.

[7] Recall value for a protein family is defined as the ratio of correctly identified proteins (which are members of the family) over the total number of family members present in the test set. For precision, the denominator is taken as the total number of proteins (both positive as well as negative test examples) present in the test which are identified as members of the family. F-measure is defined as the harmonic mean of recall and precision

4.3 Experimental Results

The results of the experimental evaluation are shown in figures 9, 10, 11, 12, 13 and 14.

Figure 9 shows the precision performance for each of the protein families. They range from 86% (for Ribonuclease T1) to 92% (for Ribonuclease A). Figure 12 shows the precision performance for each sub-families of Nu:-His-Elec catalytic triad.

Figure 10 shows the recall performance for each of the protein families. They range from 88% (for Ribonuclease T1 and Nu:-His-Elec catalytic triad) to 90% (for Ribonuclease A and Eukaryotic Lysozyme). Figure 13 shows the recall performance for the subfamilies of Nu:-His-Elec catalytic triad.

We also calculated f-measure for each of the families. Figure 11 shows the f-measure for each of the families. It ranges from 87% (for Ribonuclease T1) to 91% (for Ribonuclease A). Figure 14 shows the f-measure performance for the sub-families of Nu:-His-Elec catalytic triad.

4.4 Discussion

The experimental performance suggests that PHMM-based methods described in this paper for determining similarity of active sites works well in practice. The PHMM constructed for each family exhibits reasonably high recall, precision and f-measure values. A high degree of shared features by family members results in higher performance metrics. For instance the active sites of Ribonuclease A shares many atom types along with their geometric configuration. This is reflected by its high recall, precision and f-measures (90%, 92%, 91%). On the other hand the low degree of shared features observed in Ribonuclease T1 has translated into low recall, precision and f-measure values (86%, 88%, 87%).

.

5 Related Work

We review here computational tools and techniques related to the problem of determining similarity of active sites.

On the tool front the best known system is SPASM [10, 22]. It takes the pair $\langle$ protein structure, target active site $\rangle$ as the input and finds substructures in the protein that are similar to the active site. As we had discussed earlier comparing a substructure to an active site independently of other members of the active site's family fails to exploit the commonality amongst them. Consequently, it can fail to establish similarity with some family members, especially remote ones. A profile based approach as is done in the paper addresses this problem since profiles can capture common features of family members.

The idea of profiling active sites was first explored in the context of building the PROCAT database [20, 26], in which the term "functional template" was used for what we refer to as the active site profile in this paper. In PROCAT, functional templates are manually defined for several enzyme families. For example, it includes templates for Ribonuclease A and the five subfamilies of Histidine-based catalytic triad (see Tables 2 and 3). These templates consist of only a subset of atoms in the active site residues. For instance, only the O^{γ} atom is included in the template for the Ser-His-Asp subfamily. The decision of which atoms to include is done manually through close inspection of the structures and functional mechanisms of all the proteins in the family. The template so constructed captures the features shared by the family members. Th problem here is that template construction is a manual process thereby limiting scalability. In contrast our approach to "learn the templates" is highly automated.

A more recent work is Catalytic Site Atlas (CSA) [19, 27], a database documenting enzyme active sites and catalytic residues present in enzymes with 3-D structures. The active sites are labeled either original or derived. The former are extracted from scientific literature while the latter are associated with proteins whose primary sequences are homologous to the primary sequences of proteins containing the original active sites. An original active site and all of its derived sites constitutes a family. Templates with shared features are again constructed manually for each family.

MultiBind is yet another recent work that takes a set of active sites and automatically aligns all of them [18, 28]. The multiple alignment reveals what are the subset of atoms that are conserved among all the active sites in the set. Firstly, this approach is not statistical unlike ours. But the more important difference is that multiple alignment alone does not provide any quantitative measure of how close an active site is to the aligned sites. Without such measures it is not possible to algorithmically deduce similarity.

PHMMs were used for profiling entire protein structures in [1]. The 3-D structure is serialised into a sequence of 3-D coordinates. In other words this work uses only one geometric feature. Such an approach is useful for determining similarity of entire protein structures whose superposition has the lowest RMSD value. As we had discussed earlier (see Section 3.1) 3-D coordinates alone may not adequately capture the salient shared features of the family. Good superpositions in terms of low RMSD values may produce incompatible atom types at the superposed positions. Factoring in both physico-chemical and geometric features as is done in our approach can result in more accurate determinations of similarity and our experimental results seem to validate this hypothesis. Futhermore, as discussed below, there is no correlation between similarity of entire structures as is done in this work and active site similarity.

Finally, we remark that protein functions can also be predicted based on sequence homology or overall structure similarity. However it has been observed that there is no significant correlation between conservation of sequences, structures and active sites [11]. Hence function prediction by detection of substructures in proteins that are similar to active sites of proteins with known functions complement those based on sequence homology and structural similarity methods.

6 Conclusion

In this paper we described computational techniques for statistically profiling active sites in proteins. Specifically we adapted the successful PHMM based approach for analysis of linear sequences to encode the profiles of 3-D active sites belonging to a family. Our preliminary experience with a prototype implementation of our approach indicates that it is effective in practice.

There are several avenues for future work along the lines pursued in this paper. In our experimentation, we only utilized one geometric feature, namely distance to the center of mass. This is a relatively coarse measure. It should be possible to incorporate other geometric features as well, such as pair-wise distances between atoms. We can also incorporate additional physico-chemical features such as stereochemical and charge constraints of the active sites. Adding these features will yield richer profiles and may further improve the accuracy of prediction.

Another major idea is to depart from the linear structure of HMMs. Transitions in HMMs depend only on the previous state. While HMMs are appropriate for modeling primary structures of proteins, active sites are 3-D structures and a state transition is necessarily influenced by a set of neighboring states. Linearizing 3-D structures fails to capture such dependencies between neighboring states. Hidden Markov Random Fields (HMRF) [12] relax this limitation. HMRFs operate over undirected state graphs. The probability distribution of a random variable associated with a state in a HMRF is a function of the states in its neighborhood as defined by the graph structure. It appears that HMRFs offer a natural computing model for profiling active sites. Estimating HMRF parameters for this problem is a promising research direction.

References

[1] V. Alexandrov and M. Gerstein. Using 3d hidden markov models that explicitly represent spatial coordinates to model and compare protein structures. *BMC Bioinformatics*, 5, 2004.

[2] L. Baum. An inequality and associated maximization technique in statistical estimation of probabilistic functions of a markov process. *Inequalities*, 3:1–8, 1972.

[3] V. Borkar, K. Deshmukh, and S. Sarawagi. Automatic text segmentation for extracting structured records. In *ACM SIGMOD International Conf. on Management of Data*, 2001.

[4] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood from incomplete data via the em algorithm. *J. Roy. Stat. Soc.*, 39:1–38, 1977.

[5] S. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge University Press, 1998.

[6] S. R. Eddy. Profile hidden markov models. *Bioinformatics*, 14:755–763, 1998.

[7] R. Hughey and A. Krogh. Hidden markov models for sequence analysis: Extension and analysis of the basic method. *Comput. Appl. Biosci.*, 12:95–107, 1996.

[8] K. Karplus, C. Barrett, and R. Hugher. Hidden markov models for detecting remote protein homologies. *Bioinformatics*, 14:846–856, 1999.

[9] M. Kifer, I. Ramakrishnan, A. Ramanathan, C. Zhao, S. Jayaraman, and S. Swaminathan. Tkb: Toxin knowledge base for discovering bio-engineered threats. In *ISMB 2005*, 2005. Tool Demo and Poster.

[10] G. Kleywegt. Recognition of spatial motifs in protein structures. *J. Mol. Biol.*, 285:1887–1897, 1999.

[11] D. Korkin, F. Davis, and A. Sali. localization of protein-binding sites within families of proteins. *Protein Science*, 14:2350–2360, 2005.

[12] H. Kuensch, S. Geman, and A. Kehagias. Hidden markov random fields. *Annals of Applied Probability*, 5:577–602, 1995.

[13] A. Laurie and R. Jackson. Q-sitefinder: an energy-based method for the prediction of protein-ligand binding sites. *Bioinformatics*, 21(9):1908–1916, 2005.

[14] F. Melo and E. Feytmans. Novel knowledge-based mean force potential at atomic level. *J. Mol. Biol.*, 267:207–222, 1997.

[15] S. Mukherjee, C. Zhao, and I. Ramakrishnan. Profiling protein families from partially aligned sequences. In *SIAM Conference on Data Mining*, 2006.

[16] J. Park, K. Karplus, C. Barrett, R. Hugher, D. Haussler, T. Hubbard, and C. Chothia. Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise methods. *J. Mol. Biol.*, 284:1201–1210, 1998.

[17] L. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of IEEE*, 77(2), 1989.

[18] M. Shatsky, A. Shulman-Peleg, R. Nussinov, and H. Wolfson. recognition of binding patterns common to a set of protein structures. In *RECOMB*, pages 440–455, 2005.

[19] J. Torrance, G. Bartlett, C. Porter, and J. Thornton. Using a library of structural templates to recognize catalytic sites and explore their evolution in homologous families. *J. Mol. Biol.*, 347:565–581, 2005.

[20] A. Wallace, N. Borkakoti, and J. Thornton. Tess: A geometric hashing algorithm for deriving 3d coordinate templates for searching structural databases. application to enzyme active sites. *Protein Science*, 6:2308–2323, 1997.

[21] http://www.all.cs.sunysb.edu.
[22] http://alpha2.bmc.uu.se/usf/spasm.html.
[23] http://www.chemcomp.com/journal/sitefind.htm.
[24] http://www.expasy.org/sprot/.
[25] http://pfam.cgb.ki.se/help/scores.html.
[26] http://www.biochem.ucl.ac.uk/bsm/PROCAT/PROCAT.html.
[27] http://www.ebi.ac.uk/thornton-srv/databases/CSA/.
[28] http://bioinfo3d.cs.tau.ac.il/MultiBind/.

Profiling Protein Families from Partially Aligned Sequences*

Saikat Mukherjee
Siemens Corporate Research
saikat.mukherjee@siemens.com

Chang Zhao I.V. Ramakrishnan
Stony Brook University
{changz,ram}@cs.sunysb.edu

Abstract

Profile Hidden Markov Models (PHMMs) are recognized as powerful computational vehicles for homology search of protein sequences. Extant PHMM training approaches either use completely unaligned or aligned sequences. The PHMMs resulting from these two training approaches present contrasting tradeoffs w.r.t. alignment information and the accuracy of the search outcome. This paper describes a PHMM based technique for modeling protein families from partially aligned sequences. By exploiting the observation that partially aligned sequences give rise to independent subsequences, PHMMs corresponding to these subsequences are composed to build PHMMs for the entire sequences. An interesting aspect of the technique is that it gives rise to a family of PHMMs which are parameterized w.r.t. the alignment information. We present experimental comparison of the performance of our technique against several state of the art homology detection methods.

1 Introduction

The success of genomic work on various species has resulted in an enormous multitude of biological sequence information. This has created a rich research area centered around the development of automated techniques for analysis of these sequences. An effective means of understanding the characteristics of a new biological polymer from its sequence is through homology – whereby the sequence is compared to other similar sequences with known rich biological information. Profile hidden Markov Model[2, 5] has been proven to be able to detect remote homology.

The two dominant approaches for training PHMMs differ mainly in the way training sequences are utilized. At one extreme is training from *completely aligned* sequences where all the residues in every sequence are mapped to a column representation taking into account insertions and deletions. In contrast, (inexpensive) training from *completely unaligned* sequences uses no such information. Not surprisingly, PHMMs trained from completely aligned sequences (which we will refer to as A_PHMMs) have been shown to identify remote homologs with a much higher degree of accuracy than those trained from unaligned sequences (which we will refer to as U_PHMMs). However, producing the information about alignments is a labor intensive process involving expensive structural analysis of entire sequences. The contrasting trade-offs at the two ends of the alignment spectrum gives rise to the question: Can we develop techniques for learning profile PHMMs that trade the accuracy of remote homolog identification for alignment information? Using the notion of *partially aligned* sequences where only parts of sequences are aligned against each other, we formulate this problem as one of estimating PHMM parameters from such sequences. We will refer to PHMMs trained with such partially labeled sequences as P_PHMMs.

The essence of our approach for training PHMMs from partially aligned sequences (referred to as P_PHMM from now on) rests on the observation that a consecutive string of unaligned residues between two aligned residues can be generated only from the sequence of states lying between the match states for the aligned residues in the P_PHMM structure. Based on this observation, the algorithm decomposes P_PHMM into submodels whose parameters are separately estimated and then composed together to produce the original P_PHMM parameters. The technique is *parameterized* w.r.t. the alignment information in the sense that by varying the alignment information we can estimate the parameters of PHMMs spanning the entire spectrum from aligned PHMM at one end to unaligned PHMM (U_PHMM) at the other end.

The idea of combining PHMMS has been explored by MetaMEME[4]. However, our approach uniformly models both motif and non-motif regions as full PHMMs with match, insert, and delete states. This leads to more precise results especially when motifs do not cover significant portions of the sequences. Another closely related work is TCoffee[7] which can be used to generate a multiple alignment (from which a family model can be learned) from partial alignments.

The rest of the paper is organized as follows: In Section 2, we present algorithmic details of our technique for building P_PHMMs. Section 3 presents experimental results on the performance of P_PHMM. Section 4 concludes the paper.

*We thank Dr. Subramanyam Swaminathan for insightful discussions on this work. This research is supported by U.S. Army Medical Research Acquisition Activity Contract DAMD17-03-1-0520 and New York State Department of Health Contract C020593.

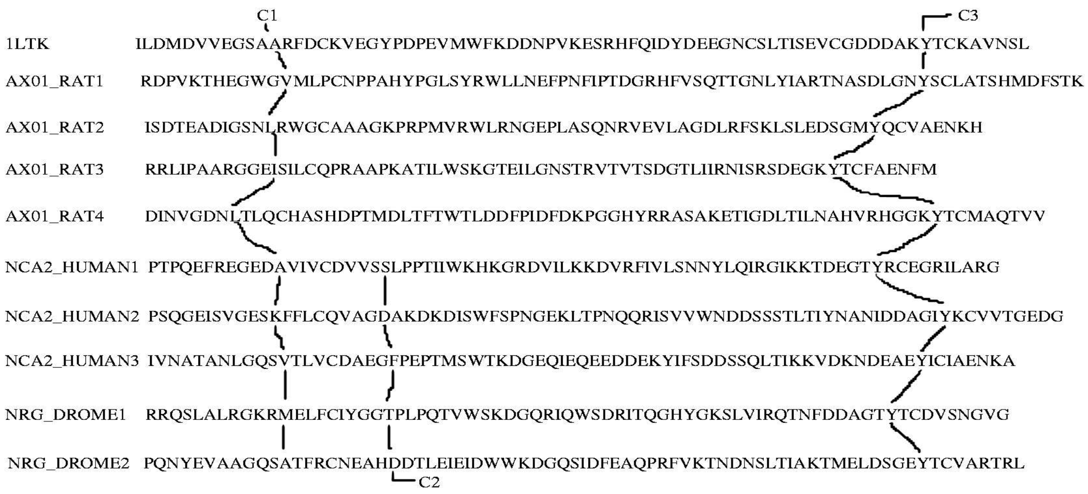

C1
C3
1LTK ILDMDVVEGSAARFDCKVEGYPDPEVMWFKDDNPVKESRHFQIDYDEEGNCSLTISEVCGDDDAKYTCKAVNSL
AX01_RAT1 RDPVKTHEGWGVMLPCNPPAHYPGLSYRWLLNEFPNFIPTDGRHFVSQTTGNLYIARTNASDLGNYSCLATSHMDFSTK
AX01_RAT2 ISDTEADIGSNLRWGCAAAGKPRPMVRWLRNGEPLASQNRVEVLAGDLRFSKLSLEDSGMYQCVAENKH
AX01_RAT3 RRLIPAARGGEISILCQPRAAPKATILWSKGTEILGNSTRVTVTSDGTLIIRNISRSDEGKYTCFAENFM
AX01_RAT4 DINVGDNLTLQCHASHDPTMDLTFTWTLDDFPIDFDKPGGHYRRASAKETIGDLTILNAHVRHGGKYTCMAQTVV
NCA2_HUMAN1 PTPQEFREGEDAVIVCDVVSSLPPTIIWKHKGRDVILKKDVRFIVLSNNYLQIRGIKKTDEGTYRCEGRILARG
NCA2_HUMAN2 PSQGEISVGESKFFLCQVAGDAKDKDISWFSPNGEKLTPNQQRISVVWNDDSSSTLTIYNANIDDAGIYKCVVTGEDG
NCA2_HUMAN3 IVNATANLGQSVTLVCDAEGFPEPTMSWTKDGEQIEQEEDDEKYIFSDDSSQLTIKKVDKNDEAEYICIAENKA
NRG_DROME1 RRQSLALRGKRMELFCIYGGTPLPQTVWSKDGQRIQWSDRITQGHYGKSLVIRQTNFDDAGTYTCDVSNGVG
NRG_DROME2 PQNYEVAAGQSATFRCNEAHDDTLEIEIDWWKDGQSIDFEAQPRFVKTNDNSLTIAKTMELDSGEYTCVARTRL
C2

Figure 1: Partial alignment information for ten ig sequences

2 Partial Alignment Profiling

Building P_PHMMs rests on the use of *partial alignment* information to *decompose* a PHMM structure into submodels and *compose* parameters computed from these submodels into the PHMM's parameters.

Partially Aligned Sequences: In a set of partially aligned sequences, alignment information is known only for a subsequence of residues in every individual sequence in the set. C_1, C_2, and C_3 in Figure 1 show three aligned columns in the ten sequences of the ig family. The alignment C_1 spans the residues $A, V, L, I, L, A, K, V, M$, and A in the ten sequences respectively and is illustrated by the leftmost solid line. Similarly, the alignment C_3 spans the Y residues in each of the ten sequences as indicated by the rightmost solid line. As illustrated in C_2, where the residues S, D, F, T, and D in only the last five sequences are aligned, it is not necessary that an alignment information has to cover all the sequences in the set. In the event of alignment being known for all the residues in every sequence, partial alignment collapses to complete alignment while total absence of any alignment information reduces to a set of unaligned sequences.

We have used the simple heuristic of taking the average length of the sequences to estimate the model length. For instance, for the ten ig family members in Figure 1, the model length computed by averaging over the size of the ten sequences is 74. By the definition of alignment, all residues aligned at a particular column are generated from the same state in the PHMM. We estimate this state by averaging over the positions of the residues, belonging to the alignment, in their corresponding sequences. For instance, for the alignment C_1 in Figure 1, the mean position where a residue in the alignment occurs in a sequence is 12. Consequently, all the ten residues in C_1 are generated from the match state M_{12}. Similarly, the ten residues in C_3 and the five residues in C_2 are generated from the match states M_{65} and M_{21} respectively.

Model Decomposition: The key to using partial alignment information for estimating PHMM parameters is the observation that a substring of unaligned residues between any two aligned residues can only be generated from the sequence of states in model positions between those corresponding to the aligned residues. For instance, in the first sequence 1LTK in Figure 1, the residues A (C_1) and Y (C_3) belong to match states M_{12} and M_{65}. The substring of unaligned symbols from R to K between the two aligned residues can be generated only from states in model positions 13 to 64 and the insert state I_{12}. This observation lets us decompose the PHMM structure into submodels where each submodel generates substrings from the original sequence. In what follows, we have *ignored gaps* in alignment information for simplicity of exposition of our technique.

In our decomposition framework, aligned residues are generated from singleton match states while substrings of unaligned residues are generated from PHMMs consisting of states in sequences of consecutive positions in the original model. We construct these PHMMs, or submodels, from the appropriate states in the original model and add begin and end states to complete the submodel structure. The PHMM P_1 in Figure 2 illustrates an example submodel. During decomposition, for a sequence with aligned residues α_n, α_m generated at match states M_i, M_j respectively and with the intermediate unaligned substring $\alpha_{n+1} \cdots \alpha_{m\ 1}$ generated from the submodel P, transitions are created from M_i to P and from P to M_j. In the event of consecutive aligned residues (i.e. $\alpha_m = \alpha_{n+1}$), the submodel P

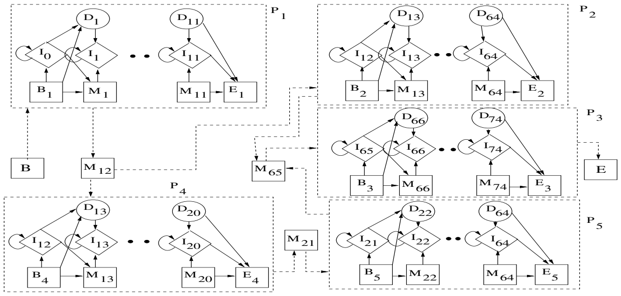


Figure 2: Decomposition of a 74 length PHMM structure using the partially aligned sequences in Figure 1

does not exist and M_i directly transitions to M_j. Figure 2 illustrates the decomposition on a PHMM structure with model length 74 using the ten partially aligned sequences of the ig family in Figure 1.

Parameter Composition: The essence of our composition technique is to estimate the original PHMM parameters from expectations of transition and emission events computed from submodels and singleton match states.

Recall that a submodel generates a set of unaligned residue substrings. For instance, the submodel P_1 in Figure 2 generates the first eleven residues of all the ten ig family sequences shown in Figure 1. This allows individual submodel parameters to be estimated by Baum-Welch training.

The singleton match states corresponding to aligned columns generate a set of residues. For instance, in Figure 2, M_{65} emits only the Y residue while M_{12} emits the residues A, V, L, I, K, and M. The emission probabilities of residues in these match states are estimated by smoothed maximum likelihood frequency counting. Also, transition probabilities between submodels and neighboring match states and vice-versa are estimated using a smoothed maximum likelihood approach. For instance, in Figure 2, if n_{M_{12},P_2} and n_{M_{12},P_4} denote the number of sequences where transitions from M_{12} to P_2 and from M_{12} to P_4 occur respectively, the probability of transition from M_{12} to P_2, $p_{M_{12},P}$, is computed as $\frac{n_{M_{12},P_2}+1}{n_{M_{12},P_2}+n_{M_{12},P_4}+2}$.

The partial alignment information in a sequence can be such that:

1. Alignment occurs at the match states M_k and M_l, where $k < i$ and $j < l$, for the residues α_n and α_m respectively.
2. Alignment occurs at the match state M_i for the residue α_n but not at M_{i+1}.
3. Alignment occurs at the match state M_{i+1} for the residue α_m but not at M_i (the converse of the above).
4. Alignment occurs at both M_i and M_{i+1} for residues α_n and α_{n+1} respectively.

Given a sequence, these four scenarios influence the computation of transition expectations for the three kinds of states in a PHMM. Let A_{S_1,S_2} denote the transition expectation between state S_1 and S_2. Apparently scenario 4 only contributes to $A_{M_i,M_{i+1}}$ which in this case is just the count of the number of times a transition is made between the singleton match states M_i and M_{i+1}. For the other three scenarios, Table 1 summarizes how the transition expectations are estimated.

Scenario	1	2	3
$A_{D_i,D_{i+1}}$	BW	N/A	N/A
A_{D_i,I_i}	BW	N/A	BW
$A_{D_i,M_{i+1}}$	BW	N/A	$A^P_{D_i,E_P} \times p_{P,M_{i+1}}$
A_{I_i,I_i}	BW	N/A	N/A
$A_{I_i,D_{i+1}}$	BW	N/A	N/A
$A_{I_i,M_{i+1}}$	BW	N/A	$A^P_{I_i,E_P} \times p_{P,M_{i+1}}$
A_{M_i,I_i}	BW	$p_{M_i,P} \times A^P_{B_P,I_i}$	BW
$A_{M_i,D_{i+1}}$	BW	$p_{M_i,P} \times A^P_{B_P,D_{i+1}}$	N/A
$A_{M_i,M_{i+1}}$	BW	$p_{M_i,P} \times A^P_{B_P,M_{i+1}}$	$A^P_{M_i,E_P} \times p_{P,M_{i+1}}$

Table 1: Transition Expectations

In Table 1, all entries marked by 'BW' means that the expectation is estimated from Baum-Welch on the appropriate submodel. For example, the expectation $A_{D_i,D_{i+1}}$ from delete state D_i to D_{i+1} for scenario 1 is given by $\sum_{t=n+1}^{t=m-1} \xi_t(D_i, D_{i+1})$.

Scenario 2 and 3 require considering a neighboring singleton match state. Let us work out scenario 3 for $A_{D_i,M_{i+1}}$. In such a situation, D_i makes a transition to the end state E_P of the submodel P which generates the unaligned substring preceding the aligned residue in M_{i+1}. Thus $A_{D_i,M_{i+1}}$ is estimated as $A^P_{D_i,E_P} \times p_{P,M_{i+1}}$, where $p_{P,M_{i+1}}$ is the probability of transition between P and M_{i+1}.

Finally, the sum of the expectations for any event over all the sequences are used to estimate its probability using a smoothed maximum likelihood technique. For instance, the probability of transition between M_i, M_{i+1} is given by:

$$p_{M_i,M_{i+1}} = \frac{\sum A_{M_i,M_{i+1}}+1}{\sum A_{M_i,M_{i+1}}+\sum A_{M_i,I_i}+\sum A_{M_i,D_{i+1}}+3},$$

where the summation denotes the cumulative value of the expectation over all the sequences.

Emission expectations of residues in states are estimated from submodels, by Baum-Welch, and from singleton match states by frequency counting. Smoothed maximum likelihood is used to compute the emission probabilities from these expectations.

3 Experimental Results

Experiments were conducted to compare the performance of P_PHMM against vanilla PHMM (U_PHMM), SAM which is a state of the art PHMM tool, an advanced multiple alignment tool TCoffee, and metaMEME. The effect of varying training set size as

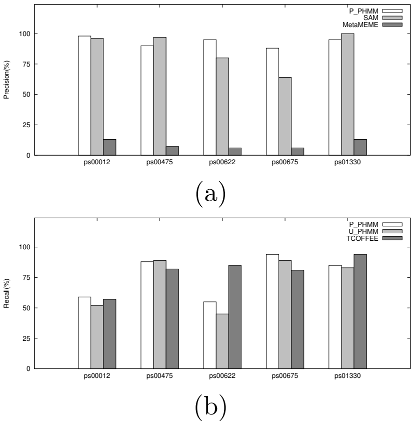
ID	M	P_PHMM		U_PHMM		SAM		TCOF.		MME.		RE	
		T	F	T	F	T	F	T	F	T	F	T	F
ps00012	221	130	3	115	0	165	6	127	2	139	967	169	143
ps00475	80	70	8	71	6	72	2	66	0	60	851	57	11
ps00622	100	55	3	45	0	85	21	85	0	84	1225	77	4
ps00675	91	86	11	81	19	86	49	73	1	86	1412	70	138
ps01330	96	82	5	80	2	90	0	90	0	24	164	59	0

(a)

ID	Members	Aligned Cols
a.1.1.2	60	186
b.1.1.2	59	122
b.34.2.1	26	176
c.47.1.5	31	101
d.169.1.1	28	173

(b)

Figure 3: (a)Experimental data with 15% training set on the 5 Prosite families (b)The 5 SCOP families



(a)

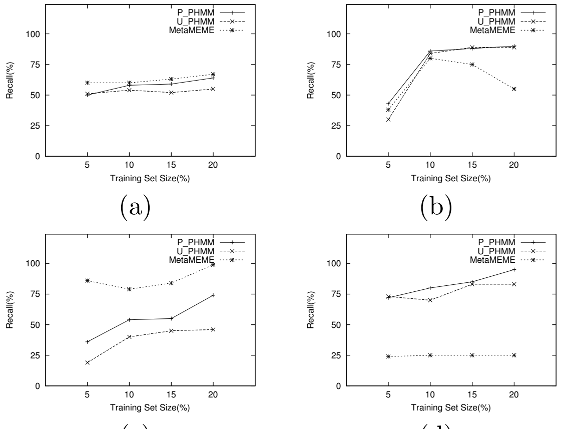
(b)

Figure 4: (a) P_PHMM precision against SAM and metaMEME (b) P_PHMM recall against U_PHMM and TCoffee

well as alignment information on the performance were also investigated.

Datasets: Regular expression based signature information available for families in the PROSITE database[3] were used to generate partially aligned sequences. Matches of a family's signature in sequences which belong to it constitute the partial alignment information for these sequences. To demonstrate the effectiveness of our technique in homology identification, 5 PROSITE families, each having at least 50 members, were chosen where RE-based pattern signatures were not very effective in identifying family members. The first column in Figure 3(a) shows the families used while the second column shows the number of members of each family in the Swiss-Prot database [1]. The models trained with P_PHMM, U_PHMM, SAM, and TCoffee were used with hmmsearch of HMMER [2] to detect homologs in Swiss-Prot while for metaMEME its own search tool, mhmms, was used. Default cutoff values were used in both the cases.

Recall and Precision: Figure 3(a) tabulates the results of the experiments for the five models on the five families using 15% of the members of each family as the training set. The columns T and F for each model reflect the number of true and false positives respectively in the test set. Figure 4 summarizes the results w.r.t. recall and precision. Observe from Figure 4(a) that the precision of P_PHMM is significantly better than metaMEME for all the families. The recall of P_PHMM is better than metaMEME for 3 of the 5 families as shown in Figure 3(a). The precision of P_PHMM is significantly better than SAM for ps00622 and ps00675 while being comparable for the other 3 families. Figure 4(b) illustrates the recall of P_PHMM against U_PHMM and TCoffee. P_PHMM has better recall for 4 of the families compared to U_PHMM and, apart from ps00622, has better or similar recall compared to TCoffee.



(a)

(b)

(c)

(d)

Figure 5: Comparing recall of P_PHMM with U_PHMM and metaMEME for (a) ps00012, (b) ps00475, (c) ps00622, and (d) ps01330

Effect of varying training set: A desirable property of any supervised learning algorithm is the improvement in performance with increased training. Figures 5 shows the change in recall with increasing training set size for P_PHMM, U_PHMM, and metaMEME. Observe that for all the four families the recall of P_PHMM increases with training set size. In contrast, vanilla PHMM or U_PHMM does not always show a increase as evident in ps00012 and ps01330. This is even more true for metaMEME which, in spite of having similar recall numbers as P_PHMM in Figure 3(a), does not demonstrate better performance with more training.

Effect of varying alignment information (SCOP): The parameterized nature of the P_PHMM algorithm was borne out by experiments conducted on families from the SCOP [6] database. The SCOP database

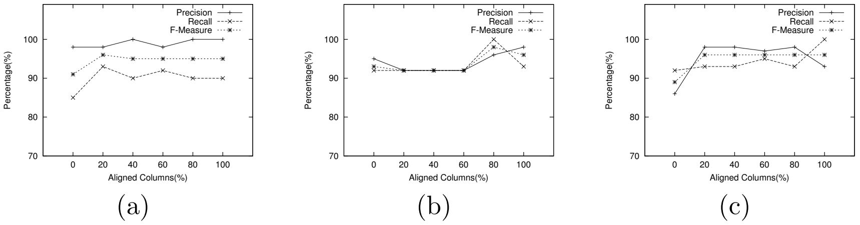


Figure 6: Impact on P_PHMM performance of varying alignment information for the SCOP families (a) a1.1.2, (b) b1.1.2, and (c) b.34.2.1

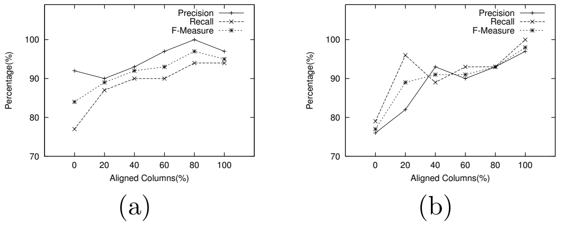


Figure 7: Impact on P_PHMM performance of varying alignment information for the SCOP families (a) c.47.1.5, and (b) d.169.1.1

provides detailed and comprehensive description of the structural and evolutionary relationship between proteins. For the purposes of our experiments, we selected 5 SCOP families each having at least 25 members. Multiple alignment of these families was derived from the PALI [8] database which provides alignments of proteins in the SCOP database. Column 1 in Figure 3(b) lists the ids of these 5 families, while Columns 2 and 3 show the number of family members and the number of aligned columns in their multiple alignment.

P_PHMMs were trained for each of these 5 families. The training set size, for each family, was fixed at a randomly chosen set of 25% of its total members. The amount of alignment information was successively varied from the use of 0% (completely unaligned), to 20%, 40%, 60%, and 80% of the number of aligned columns in the multiple alignment of the family. The test set for each of these families consisted of all the 5179 domains from all the 1029 families in PALI release 2.3. Recall, precision, and F-measure of homology detection were calculated for them. Figure 6 and Figure 7 graphically illustrates the impact on the three metrics with varying alignment information on all the 5 families. While all the 5 families show increase in the values of the three metrics with alignment information, this is especially perceptible in the SCOP families *c*.47.1.5 and *d*.169.1.1 in Figure 7(a) and (b) respectively.

4 Discussions

In this paper, we proposed a parameterized technique for learning PHMMs from partially aligned sequences. Our technique was based upon decomposing a PHMM structure into submodels and composing these submodels' parameters into that of the PHMM.

Usually, PHMM parameters are learned with the Baum-Welch algorithm from unaligned sequences. Note that it is non-trivial to modify Baum-Welch to handle partial alignment information. Baum-Welch is defined in terms of a pair of algorithms which are formulated in a greedy, recursive manner without any lookahead capability. Consequently, incorporating alignment information at a current position for residues which occur after it in the sequence is difficult. Considering such information is necessary to restrict the assignment of expectation values to valid states only. Incorporating other sources of partial alignment information, such as PSSMs, into our framework is a topic worth exploring.

References

[1] A. Bairoch and B. Boeckmann. The swiss-prot protein sequence data bank. *Nucleic Acids Res.*, 20:2019–2022, 1992.

[2] S. R. Eddy. Profile hidden markov models. *Bioinformatics*, 14:755–763, 1998.

[3] L. Falquet, M. Pagni, P. Bucher, N. Hulo, C. Sigrist, K. Hofmann, and A. Bairoch. The prosite database, its status in 2002. *Nucleic Acids Res.*, 30:235–238, 2002.

[4] W. Grundy, T. Bailey, C. Elkan, and M. Baker. Meta-meme: Motif-based hidden markov models of protein families. *Computer Applications in Biosciences*, 13(4):397–406, 1997.

[5] K. Karplus, C. Barrett, and R. Hugher. Hidden markov models for detecting remote protein homologies. *Bioinformatics*, 14:846–856, 1999.

[6] A. Murzin, S. Brenner, T. Hubbard, and C. Chothia. Scop: A structural classification of proteins database for the investigation of sequences and structures. *J. Mol. Biol.*, 247:536–540, 1995.

[7] C. Notredame, D. Higgins, and J. Heringa. T-coffee: A novel method for fast and accurate multiple sequence alignment. *J. Mol. Biol.*, 302:205–217, 2000.

[8] S.Balaji, S. Sujatha, S. Kumar, and N. Srinivasan. Pali-a database of alignments and phylogeny of homologous protein structures. *Nucleic Acids Res.*, 29:61–65, 2001.